

2'nd
Edition

CCNA 200-301 v1.1

By: Mahmoud Tarek

<https://www.linkedin.com/in/mahmoudtarek/>

Table of Contents:

1. Network Fundamentals	15
1.1. Network Components.....	15
1.2. Network Topologies.....	16
1.3. Data transportation methods (Address Grouping).....	20
1.4. Collision Domain and Broadcast Domain.....	21
1.5. Types of Networks.....	23
1.6. Signal Transmission Modes.....	23
1.7. Network Reliability.....	24
2. Network Model	26
2.1. Standards Organizations.....	26
2.2. OSI Model.....	28
2.3. Data Encapsulation Process.....	30
2.4. OSI Layers.....	31
2.4.1. Application layer.....	31
2.4.2. Presentation Layer:.....	31
2.4.3. Session Layer.....	32
2.4.4. Transport layer.....	32
2.4.5. Network Layer.....	38
2.4.6. Data Link Layer.....	39
2.4.6.1. Ethernet Frame.....	39
2.4.7. Physical layer.....	42
2.5. Application Layer Services Overview.....	43
2.6. Port Number.....	45
2.7. Application Layer Services.....	46
2.7.1. HTTP.....	46
2.7.2. FTP.....	48
2.7.3. SMTP.....	48
2.7.4. Telnet.....	49
2.7.5. DHCP.....	49
2.7.6. SMP & NFS.....	53
2.7.7. P2P (Peer to peer).....	53
2.7.8. DNS.....	54
3. IPv4	59

3.1. IP Address Classes.....	60
3.2. Private IP Range.....	61
3.3. Ping.....	62
3.3.1. Ping Command Responses.....	62
3.3.2. Ping Command Characteristics.....	62
3.4. IPv4 Subnetting.....	63
3.4.1. Fixed Subnetting.....	63
3.4.2. Variable Length Subnet Mask (VLSM).....	65
3.4.3. ARP.....	67
3.5. IPv4 Header.....	69
4. IPv6.....	71
4.1. IPv6 Abbreviations.....	71
4.2. Two methods to get IPv6.....	72
4.3. EUI-64 (Link Local Address or LLA).....	73
4.4. IPv6 Zone Identifier.....	75
4.5. SLAAC.....	77
4.6. IPv6 Addressing.....	78
4.7. Address Grouping.....	79
4.8. IPv6 Header.....	79
5. Cables.....	81
5.1. Coaxial Cable.....	81
5.2. Twisted Pair Cables.....	83
5.2.1. Unshielded Twisted Pair - UTP.....	83
5.2.2. Shielded Twisted Pair - STP.....	83
5.2.3. Clipping Types.....	86
5.3. Fiber Optic Cables.....	91
6. Configuration Modes.....	94
6.1. Configuration Types.....	94
6.2. Configuration Modes.....	98
7. Router Passwords.....	105
7.1. Password Encryption.....	106
7.2. Privileges Levels.....	108
8. Router Components.....	112
8.1. Router Interfaces.....	114

8.2. Router Memories.....	116
8.3. Router Boot Sequence.....	117
8.4. Save Configurations.....	118
9. Telnet & SSH.....	120
9.1. Telnet.....	120
9.2. Secure Shell - SSH.....	123
9.3. SSH Configuration.....	123
10. Banner.....	125
10.1. MOTD.....	125
10.2. Login Banner.....	125
11. Filtration.....	126
12. Router Password Recovery.....	128
13. CDP - CISCO Discovery Protocol.....	133
13.1. LLDP - Link Layer Discovery Protocol.....	138
14. Routing.....	139
14.1. Static Routing.....	139
14.1.1. Static Routing Lab.....	141
14.1.1.1. Default Routing.....	147
14.1.1.2. Loopback Interface.....	150
14.2. Dynamic Routing.....	151
14.2.1. Types of Dynamic Routing Protocols.....	151
14.2.2. Distance Vector Routing Protocols (AD & Metric).....	154
14.3. Floating Static Route.....	157
15. RIP.....	160
15.1. RIP Stages.....	160
15.1.1. At Startup.....	161
15.1.2. At Change.....	164
15.1.3. At Convergence.....	165
15.2. Split Horizon And Holddown Timer.....	166
15.3. RIPv1 vs RIPv2.....	167
15.4. RIP Futures.....	169
15.5. RIP Configuration.....	171
15.5.1. RIP Summarization.....	174
15.5.2. RIP Authentication.....	175

15.5.3. RIP Equal Cost Load Balance.....	177
15.5.4. RIP Troubleshooting:.....	178
16. OSPF.....	179
16.1. OSPF Stages.....	179
16.1.1. At Startup.....	179
16.1.1.1. Neighbor Discovery.....	179
16.1.1.2. Database Exchange.....	182
16.1.2. At Convergence.....	185
16.1.3. At Change.....	185
16.2. OSPF Versions.....	185
16.2. OSPF Network Types.....	186
16.3. DR and BDR.....	188
16.4. OSPF Areas.....	189
16.5. Area and LSA Types.....	191
16.6. OSPF Lab-1.....	193
16.6.1. OSPF Authentication.....	198
16.6.2. OSPF Lab-2.....	199
16.7. Default Information Originate.....	202
16.8. Interarea Route Summarization.....	203
17. Serial Interfaces.....	205
18. DHCPv4.....	210
18.1. Reservation.....	215
18.2. IP-Helper Address.....	217
19. DHCPv6.....	219
19.1. Static IPv6 Configuration.....	219
19.2. Enable IPv6 Routing.....	220
19.3. Configure SLAAC.....	221
19.4. Configure Stateless DHCPv6 with SLAAC.....	223
19.5. Configure Stateful DHCPv6.....	225
20. Access Control List.....	227
20.1. Access-List LAB.....	228
20.1.1. Configure Routing.....	228
20.1.2. Standard ACL Configuration.....	230
20.1.3. Extended ACL Configuration.....	236

20.1.4. Named ACL.....	240
20.1.5. Access-Class.....	243
20.1.6. Filter debugging using ACL.....	244
20.1.7. IPv6.....	244
20.2. Notes.....	245
21. NAT.....	246
21.1. NAT Types.....	247
21.2. How PAT Work.....	248
21.3. LAB Configuration.....	250
21.3.1. Static NAT.....	253
21.3.2. Dynamic NAT.....	257
21.3.3. PAT.....	258
22. Router switching Modes.....	259
22.1. How Router Builds the Routing Table.....	259
22.2. Router Switching.....	260
23. Switching Essentials.....	263
23.0.1. Switching Modes.....	264
24. IOS License.....	266
24.1. CISCO IOS Naming Conventions.....	266
24.2. CISCO IOS System Image Packages.....	269
24.3. IOS 15 System Image Packages.....	270
24.4. Managing CISCO IOS.....	271
24.5. Software License.....	272
25. Virtual LAN - VLAN.....	277
25.1. Vlan ID.....	278
25.2. VLAN Ranges.....	279
25.3. Types of Vlan Membership.....	280
25.4. Types of Vlan.....	282
25.5. Vlan Configuration.....	284
25.6. Vlan Port Types.....	287
25.6.1. Access Port.....	287
25.6.2. Trunk Port.....	287
25.7. DTP - Dynamic Trunking Protocol.....	288
25.8. Vlan Trunking Protocols.....	291

25.9. Allowed Vlans.....	293
25.10. Voice vlan configuration.....	294
25.11. Vlan LAB.....	298
25.12. Vlan Hopping.....	299
25.12.1. Switch Spoofing (aka Switch Masquerading).....	299
25.12.2. Double Tagging Attack.....	302
26. Spanning Tree Protocol - STP.....	305
26.1. Switch Functions Preview.....	305
26.2. Switching Loop.....	307
26.3. How STP Work.....	310
26.3.1. Example 1.....	311
26.3.2. Example 2.....	317
26.3.3. Spanning Tree Port States.....	319
26.3.4. BPDU Timers.....	320
26.4. Per VLAN Spanning Tree- PVST.....	321
26.5. STP Toolkit.....	325
26.5.1. PortFast.....	325
26.5.2. BPDU Guard.....	328
26.5.3. BPDU Filter.....	331
26.5.4. Root Guard.....	333
26.5.5. Loop Guard.....	337
26.6. STP Versions.....	341
26.7. Rapid Spanning-Tree Protocol (RSTP).....	342
27. Layer 2 Security Futures.....	348
27.1. Port Security.....	348
27.1.1. Port Security Violation Modes.....	353
27.1.2. Port Security Aging time.....	354
27.1.3. Port Security Sticky source MAC address.....	355
27.2. DHCP Snooping.....	356
27.3. Dynamic ARP Inspection.....	361
27.3.1. ARP Review.....	361
27.3.2. Gratuitous ARP.....	363
27.3.3. DAI.....	364
27.3.4. DAI Optional Checks.....	367

27.3.5. ARP ACL.....	368
28. POE.....	369
28.1. POE Terminologies.....	370
28.2. Types and Standards.....	372
28.3. Detecting Powered Devices.....	374
28.4. POE Configuration.....	374
28.5. Common Reasons Why PoE Might Fail.....	376
29. Inter VLAN Routing.....	377
29.1. Vlan routing on the router.....	377
29.1.1. Traditional inter VLAN routing.....	377
29.1.2. Router-on-a-stick.....	378
29.2. MultiLayer Switch - MLS.....	380
29.2.1. Interface Status.....	381
30. Ether Channel.....	382
30.1. Layer 2 EtherChannel.....	382
30.1.1. Layer 2 Etherchannel Configuration.....	387
30.1.2. Configuring "ON" Mode.....	389
30.2. Layer 3 Etherchannel Configuration.....	392
31. VTP.....	394
31.1. VTP Pruning.....	403
31.2. VTP Version 3.....	407
32. FHRP.....	411
32.1. HSRP.....	413
32.1.1. HSRP Configuration.....	414
32.1.2. Scenario-2.....	418
32.1.3. Scenario-3.....	420
32.1.4. HSRP Options.....	421
32.1.5. HSRP in switching.....	423
32.2. HSRPv1 VS HSRPv2.....	424
32.3. VRRP.....	425
32.4. GLBP.....	426
33. WAN Technology.....	434
33.1. Some Terms.....	435
33.2. How Router Processes The Traffic.....	435

33.3. Types of WAN Technology.....	437
33.3.1. Circuit Switching - Leased line.....	437
33.3.2. On demand Circuit Switching.....	439
33.3.3. Packet Switching.....	439
33.3.3.1. Frame relay technology overview.....	440
33.3.4. Cell Switching.....	444
33.3.5. Label Switching.....	445
33.3.5.1. How MPLS Work.....	446
33.3.5.2. MPLS Stages.....	447
33.3.5.3. MPLS Header.....	450
33.3.5.4. MPLS Applications.....	451
33.3.6. Metro Ethernet.....	453
33.3.7. Broadband.....	454
34. VPN (Virtual Private Network).....	455
34.1. VPN Objectives.....	456
34.1.1. Confidentiality.....	456
34.1.2. Authentication.....	457
34.1.3. Integrity.....	458
34.1.4. Anti reply.....	458
34.1.5. VPN Protocols.....	458
34.2. VPN Modes.....	459
34.3. How to generate tunnel (VPN Protocols).....	460
34.4. VPN Configuration types.....	461
34.5. GRE Tunnel.....	462
34.5.1. Configuration.....	463
34.6. VPN Lab 1: IPsec.....	469
34.6.1. Configuration.....	469
34.6.2. All Configuration.....	476
35. VPN lab 2 - GRE over IPsec.....	477
35.1. GRE Configuration.....	478
35.2. GRE over IPsec.....	483
35.2.1. IPSec Configuration.....	484
36. IPv6 Migration.....	485
36.1. IPv6 over GRE Tunnel (Static Tunneling).....	486

37. QoS.....	492
37.1. Why Voice take a higher priority.....	492
37.2. Buffer and Que.....	493
37.3. Characteristics of network traffic.....	494
37.4. Classification and Marking.....	494
37.4.1. Marking.....	494
37.4.2. Classification.....	495
37.5. Trust Boundaries.....	500
37.6. Congestion Avoidance.....	500
37.6.1. WTD - Weighted Tail Drop.....	502
37.6.2. WRED - Weighted Random Early Detection.....	503
37.6.3. Congestion management.....	503
37.6.3.1. FIFO - First in First Out (no congestion management).....	503
37.6.3.2. WFQ - Weighted Fair Queueing.....	503
37.6.3.3. CBWFQ (Class Based Weighted Fair Queueing).....	504
37.6.3.4. LLC - Low Latency Queueing.....	504
37.7. Scheduling features.....	505
38. NTP.....	506
38.1. Time Synchronization Importance.....	506
38.2. Stratum Value.....	507
38.3. NTP Modes.....	508
38.4. NTP Versions.....	508
38.5. NTP LAB Configuration.....	509
39. SNMP Overview.....	513
39.1. SNMP Applications.....	513
39.2. SNMP Versions.....	514
39.3. SNMP Messages.....	515
39.4. SNMP Lab.....	516
40. Info Security and Cyber Security Fundamentals.....	521
40.1. Types of Securing data.....	521
40.2. CIA Triad.....	522
40.3. Security Terminology.....	524
40.4. Risk Assessment.....	525
40.4.1. CVSS v3.0 Base Metrics.....	526

40.5. Vulnerability Categories.....	529
40.6. Common Attack Methods.....	531
40.6.1. Trojan Types.....	534
40.6.2. Merge two apps in single exe file.....	535
40.7. Threat Actors.....	537
40.8. Hacker Categories.....	538
40.9. Malware Analyzing.....	538
40.10. Evaluation Techniques Against Security Devices.....	540
40.11. Mitigation Methods.....	541
40.11.1. Network Segmentation.....	541
40.11.2. Firewall.....	542
40.11.3. IDS vs IPS.....	543
40.11.4. NIDS, NIPS Placement.....	545
40.11.5. Some Basic Router Security.....	546
40.11.6. IOS Upgrade.....	549
41. LAN Architectures.....	550
41.1. Spine-leaf Architecture.....	558
41.2. SOHO.....	559
41.3. Stackwise Technology Overview.....	560
41.4. Some terminology.....	561
42. VRF Lite.....	562
42.1. VRF Light Lab.....	564
43. WIFI Technology.....	572
43.1. Access Point Components.....	575
43.2. WiFi Topology Types.....	576
43.3. AP Operating Modes.....	581
43.4. Licensed & Unlicensed Frequency Bands.....	582
43.5. Wi-Fi Frequencies and Channels.....	582
43.6. Wi-Fi Terminology.....	584
43.7. Wi-Fi Components:.....	585
43.8. Frame Format.....	586
43.9. Association Process.....	587
43.10. Message Types.....	588
43.11. Wireless AP Deployment Methods.....	588

43.11.1. Autonomous APs.....	589
43.11.2. Lightweight APs.....	591
43.11.2.1. Lightweight AP operational modes.....	594
43.11.3. Cloud Based Architecture.....	596
43.12. Factors Affecting Wi-Fi.....	600
43.13. Wireless Network Security.....	602
43.13.1. Authentication.....	602
43.13.2. Encryption.....	608
43.13.3. Integrity.....	608
43.13.4. Encryption and Integrity Methods.....	609
43.13.5. WPA.....	610
43.14. Wireless Configuration.....	613
43.14.1. Switch Configurations.....	615
43.14.2. WLC Configuration.....	617
43.14.2.1. Initial Setup.....	617
43.14.2.2. GUI Configuration.....	622
43.14.2.3. WLC Interfaces and Ports.....	627
43.15. Packet Tracer Lab.....	637
43.15.1. WLC on GNS3.....	649
44. Virtualization and Cloudding Overview.....	652
44.1. Virtualization Overview.....	652
44.2. Benefits of Virtualization.....	653
44.3. Virtual Machine Terminology.....	654
44.4. Vmware Interfaces.....	655
44.5. Types of Hypervisors.....	656
44.5.1. VM Locations.....	657
44.6. Cloud Computing Services.....	658
44.6.1. Cloud Computing Types.....	659
44.6.2. Cloud Computing Advantages.....	659
44.6.3. Cloud Service Model.....	659
44.7. Containers.....	660
45. Intro to Network Automation.....	664
45.1. Network Automation Benefits.....	664
45.2. Network Automation Tools.....	665

45.3. Software-Defined Networks (SDN).....	665
45.3.1. Network Logical Planes.....	666
45.3.1.1. Data Plan Functions.....	666
45.3.1.2. Control Plan Functions.....	667
45.3.1.3. Management Plan Functions.....	668
45.3.2. SDN Concept.....	669
45.3.2.1. SDN Controller.....	670
45.3.2.2. SBI.....	671
45.3.2.3. NBI.....	671
45.4. Data Serialization (Jeson, XML and YAML).....	673
45.4.1. Data Serialization.....	673
45.4.2. JSON.....	674
45.4.3. XML.....	678
45.4.4. YAML.....	679
45.5. REST API.....	680
45.5.1. CRUD & HTTP.....	680
45.5.2. REST API Architecture.....	683
45.5.3. CISCO DevNet.....	685
45.5.4. Cisco Catalyst Center LAB.....	685
45.5.5. REST API Authentication.....	692
45.5.5.1. Basic Authentication.....	694
45.5.5.2. Bearer Authentication.....	695
45.5.5.3. API Key.....	696
45.5.5.4. OAuth 2.0.....	697
45.6. SDN Architecture.....	701
45.6.1. SD-Access Architecture.....	702
45.6.2. Fabric, Underlay and Overlay Concepts.....	703
45.6.3. SD-Access Deployment.....	705
45.6.4. How data forwarded in SD-Access Network.....	706
45.6.5. CISCO DNA Center.....	707
45.7. Ansible, Puppet, & Chef Overview.....	709
45.7.1. Configuration Drift.....	709
45.7.2. Configuration Management Tools.....	710
45.7.2.1. Ansible.....	711

45.7.2.2. Puppet.....	712
45.7.2.3. Chef.....	713
45.7.3. Infrastructure as Code.....	715
45.7.4. Configuration Provisioning.....	716
45.7.5. Configuration Management vs Configuration Provisioning.....	717
45.7.5.1. Mutable vs Immutable Infrastructure.....	717
45.7.5.2. Procedural vs Declarative Approaches.....	718
45.7.6. Terraform.....	719
46. AI and ML Overview.....	722
46.1. AI.....	722
46.2. ML.....	723
46.3. Types of ML.....	725
46.4. Predictive and Generative ML Models.....	730
46.4.1. Predictive AI.....	730
46.4.2. Generative AI.....	731
46.4.3. Predictive & Generative AI Applications in Networks.....	732
46.5. AI in CISCO Catalyst Center.....	736

1. Network Fundamentals

الشبكة (Network) هي مجموعة من الأجهزة (زي الـ PCs والـ Routers والـ Switches) متصلة بعضها لتقديم خدمة معينة للمستخدم، زي مثلاً مشاركة الـ Hardware زي الطابعة، أو مشاركة البيانات، ومشاركة خدمات الصوت والفيديو (VOIP - Video Conference).

1.1. Network Components

- End Devices

End Devices ➔ (PC, Printer, IP Camera, Server, ..etc)

الـ End Devices او الأجهزة الطرفية دائماً تكون في نهاية الشبكة وتكون Source او Destination اي بيانات تتبع في الشبكة و كل جهاز يكون ليه عنوان في الشبكة اسمه IP .. الأجهزة دائماً تسمى Host (يعني بتنضيف خدمة معينة)، او تسمى Client لو كانت بتحتاج خدمة من اي .Email Server متوصلاً بيه زى صفحة Web من Email او Web Server

- **Intermediate Devices - Network Devices**

Intermediate Devices ➔ (Router, Switch, Access Point, Firewall, ..etc)

الأجهزة الوسيطة هي اللي بتوصل الـ End Devices ببعض وبباقي الأجهزة في الشبكة .. مع تحديد المسار المناسب لنقل البيانات.

Intermediary Devices used to:

- Regenerate & retransmit data signals
 - Error detection
 - Direct the data to the best path
 - Permit or deny data flow according to security policies

- Media → (Copper - Optical Fiber - Wireless)

هـ الوـسـطـ الـلـىـ يـنـقـلـ الـبـيـانـاتـ مـنـ الـS~ur~eـ إـلـىـ الـD~es~t~in~at~ionـ

1.2. Network Topologies

هي طريقة توصيل الأجهزة في الشبكة وهنا لازم نعرف الفرق بين الـ **Physical Topology** وهي طريقة توصيل الأجهزة فيزيائيا، اما الـ **Logical Topology** هي طريقة تواصل الأجهزة من خلال الـ

Bus Topology ♦ مجموعة من الأجهزة متصلة مع بعضها على كابل واحد .. بيبقى في Terminal في نهاية الـ **Cables** عشان يمنع الـ Loop او ارتداد البيانات.

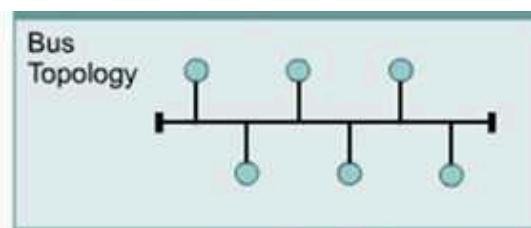
من عيوبها لو جهاز اتفرق او فصل « الشبكة كلها هتقف والعيب الثاني One send & All Receive يعني لو اتنين بعثوا داتا في نفس الوقت هيحصل تصادم او Data Collision .

مشكلة الـ **Collection** بتتحل عن طريق استخدام الـ **CSMA/CD Mechanism** .. وطريقة عمله:

◇ الجهاز اللي عايزة بيعت داتا بيعمل Carrier Sense - CS الأول على الكبل .. لو في إشارة « بىنتظر .. لو مفيش » بيعت البيانات.

◇ لو جهازين بعثوا داتا في نفس الوقت « هيحصل Collision .. لما الجهاز يكتشف ان في Collision » بيوقف ارسال البيانات وبيبعث Jam Signal لكل الـ Nodes اللي معاه عشان يستنحو وميبيعتوش داتا .. وكل جهاز بيعمل Generate لـ Random time عشان بيعت البيانات.

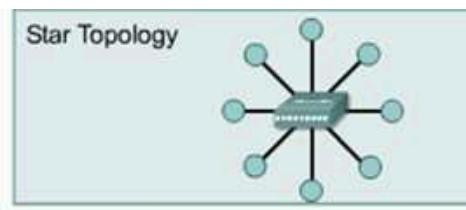
CSMA/CD → *Carrier Sense Multiple Access / Collision Detection*



Physical Topology: BUS

Logical Topology: BUS

سواء Central Device مجموعه من الاجهزه متصلة مع بعضها عن طريق Star Topology .. او Switch Hub ودي الطريقة الشائعه حاليا في الشبكات.



لو الـ Central Device عباره عن Hub .. فالـ Hub لما يستقبل Data من جهاز A مثلـا .. رايحة لجهاز B بيعملها Flood على كل الـ Interfaces ما عادا الـ Interface اللي وصلت منه.

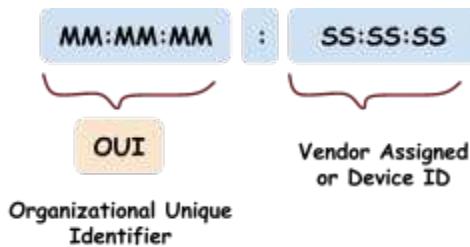
في الحالة دي ممكن يحصل Collision لو جهازين بعثوا Data في نفس الوقت .. عشان كدا بنستخدم تقنية الـ CSMA/CD .. الـ Star Physical Topology على سكل Logical Bus تكون Bus لأن في وجود الـ Hub بـها البيانات بتنتقل بنفس طريقة الـ BUS Topology وكان الاجهزه كلها على كابل واحد.

أما في حالة الـ Switch .. فالـ Switch لما يستقبل Data من جهاز A مثلـا .. رايحة لجهاز B بـها Flood لأول Frame فقط على كل الـ Interfaces اللي الـ Data اللي وصلت منه.

الـ MAC Address Table بيبني Switch داخل الـ RAM متخزن فيه الـ MAC Address والـ MAC المقابل للـ MAC دا عشان يقدر يوجه البيانات او بمعنى ادق، بيعمل Switch للبيانات يعني تبديل .. الـ MAC بيفضل متخزن لمدة 300 ثانية اسمها الـ Aging time وبعد كدا يتفسح .. في الحالة دي شكل الـ Physical Topology عباره عن Star، والـ Logical نوعها كمان Star.

نبذة عن الـ MAC Address

- الـ MAC Address اختصار لـ Media Access Control وهو عبارة رقمية يمثل في الـ Octet-6 وكل Octet عبارة عن 8 bit يعني كله عبارة عن 48 bit.
- يتقسم لجزئين: الـ OUI أول 24 بت (3 بايت) بتدل على الشركة المصنعة (زي Intel أو Cisco). والمفروض انه بيكون ثابت لكل شركة وتنانى جزء بيتغير على حسب كل NIC.



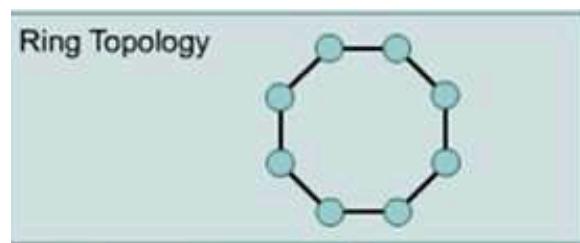
- الـ MAC Address يكون Unique على كل جهاز على مستوى العالم عن طريق منظمة الـ IEEE اي شركة مصنعة لكرات الشبكة بتشتري الـ OUI من منظمة الـ IEEE. لكن طبعا في شركات بتصنع كروت شبكة بشكل غير رسمي، وفي Softwares بتغير الـ MAC .. وبالتالي ممكن يحصل تشابه.

EUI-48 Standard MAC Address

- الـ MAC ممكن يطلق عليه CAM Content Addressable Memory ودا الاسم الـ Offical .Burned in Address - BIA بتسميه CISCO

❖ **Ring Topology** ← : مجموعة من الاجهزه متصلة مع بعضها في شكل حلقة .. مشكلتها البطء و كل

جهاز يبقى محتاج كرتين شبکه Network Interface Card - NIC.



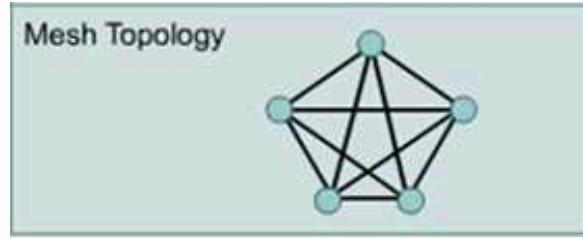
لها اسم تاني (FDDI) وشكل الا Ring عبارة عن Physical Topology وال Logical Topology عن

❖ **Mesh Topology** ← : مجموعة من الاجهزة كلها متصلة ببعضها بمعنى ان الجهاز الواحد متصل بكل

الاجهزة اللي معاه ف الشبكة ويستخدم لتوصيل switches مجموعة

High Availability للشبكة .. ويطلق عليها أيضا Redundant Topology، لأنها بتتوفر اكتر من مسار من اي

Node للوصول إلى أي Node تانية.



وفي نوع تاني اسمه Partial Mesh بنوصل فيه الاجهزة المهمة فقط بكل الاجهزة.

1.3. Data transportation methods (Address Grouping)

Data is Transported over a network by 3 methods

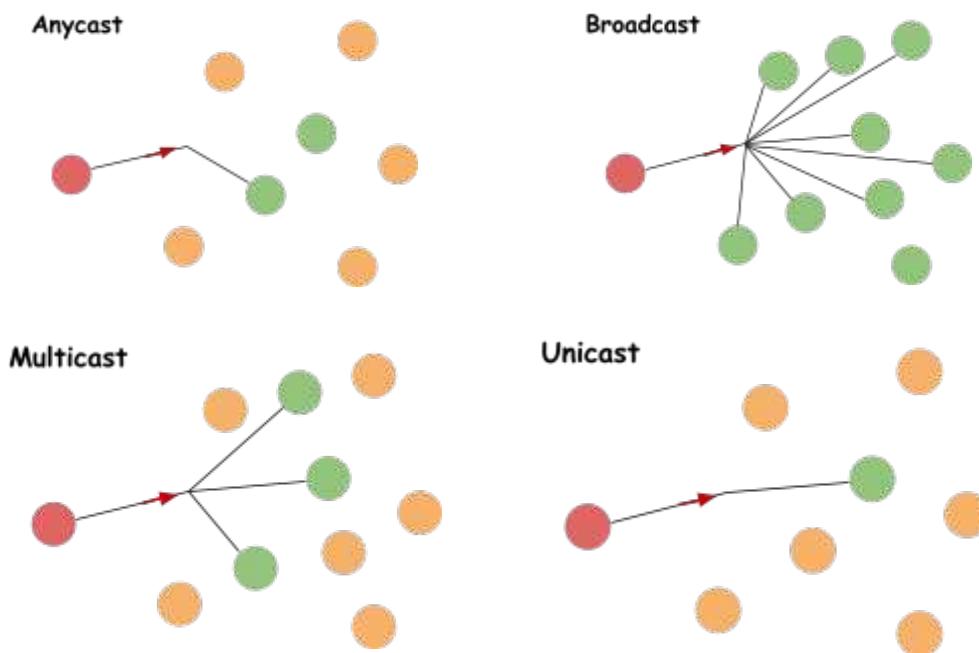
1. **Broadcast**: جهاز يرسل البيانات لكل الأجهزة اللي عنده ف الشبكة مع العلم ان الـ Broadcast هي

عملية مزعجة لازم تخلص منها.

2. **Multicast**: هو جهاز يرسل لمجموعة معينة من الأجهزة.

3. **Unicast**: جهاز يرسل لجهاز واحد فقط.

4. **Anycast**: ارسال البيانات الى اقرب جهاز على حسب قواعد معينة زي المسافة والمسار.



1.4. Collision Domain and Broadcast Domain

الـ Collision Domain والـ Broadcast Domain مفهومين أساسيين في الشبكات، بيوضحوا ازاي البيانات بتتحرك بين الأجهزة وايه اللي بيحدد نطاقها.

1. Broadcast Domain (نطاق البث):

- هو النطاق اللي بيسمع فيه رسالة Broadcast .. يعني لما جهاز بيقول "يا جماعة، مين عنده العنوان ده؟"، كل الأجهزة في النطاق ده بتسمع الكلام.
- ايه اللي بيفصل النطاق او بيقسم الـ Domains:
 - الـ Router هو اللي بيقسم الـ Broadcast Domains، لأنه مش بيعدى رسائل الـ Broadcast من شبكة لشبكة تانية.
 - الـ Switch أو الـ Hub مش بيفصلوا الـ Broadcast Domain، يعني كل الأجهزة المتصلة بيهم بتكون في نفس النطاق.
 - لو الـ Broadcast Domain كبير جداً (يعني أجهزة كتير متصلة في نفس النطاق)، الشبكة بتبطأ لأن كل جهاز بيضطر يسمع ويرد على كل رسائل الـ Broadcast.

2. Collision Domain (نطاق التصادم):

- هو مجموعة الأجهزة اللي لو اتنين فيهم بعثوا بيانات في نفس اللحظة، البيانات دي بتتصادم وتضيع. التصادم ده بيحصل لما الأجهزة بتحاول تتكلم على نفس السلك أو القناة في وقت واحد.
- مثلاً لو عندك Hub متوصّل بيـه 3 أجهزة، لو اتنين فيهم بعثوا بيانات مع بعض، البيانات الـ Hub بيعت البيانات على كل الأسلال في وقت واحد، وبالتالي هيحصل "Collision".
- الـ Switch بيفصل الـ Collision Domains لأنـه بيعت البيانات للجهاز المحدد فقط، مش بيعملها لكل الأجهزة. والراوتر برضو بيفصل الـ Collision Domains لأنـه بيقسم الشبكات أصلـاً.

زي ما قلنا ان الـ Hub يبعثوا الـ Broadcast Message لكل الاجهزه المتصلة وبالتالي كل الاجهزه تقع في نطاق واحد او Broadcast Domain واحد.

بالنسبة للـ Collision Domain على الـ Hub .. لو اي جهازين بعثوا لبعض «» هيحصل Collision، وبالتالي كل الاجهزه تقع في نطاق تصدام واحد .. اما الـ Switch بيخرن البيانات اللي بتوصله في Buffer وبعدين بيعت واحدة واحدة وبالتالي مفيش Collision يحصل على الـ Switch افتراضيا لكن كل جهاز على الـ Switch موجود في Collision Domain لوحدو.

Broadcast Domain

- For Hub: 1
- For Switch: 1
- For Router: Number of Interfaces

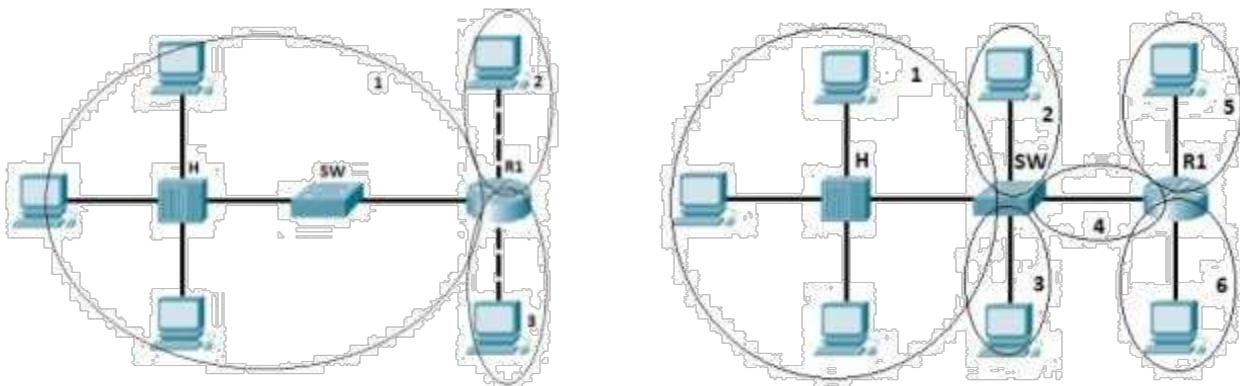
Collision Domain

- For Hub: 1
- For Switch: Number of Interfaces
- For Router: Number of Interfaces

مثال

الشبكة على اليمين فيها 3 Broadcast Domain و 3 Collision Domain والشبكة على الشمال فيها 6

Collision Domain 4 و Broadcast Domain



1.5. Types of Networks

1. **Local Area Network - LAN**: هي عبارة عن مجموعة من الأجهزة متصلة مع بعضها في مساحة محدودة زي شبكة البيت او شبكة في ساير صغير .. السرعة بتوصل فيها لـ $10Gbps$ وبنستخدم فيها بروتوكولات Gigabit Ethernet او Fast Ethernet او Giga Ethernet .

2. **Metropolitan Area Network - MAN**: و هي مجموعة من الـ LANs متصلة مع بعضها ولكنها اقل من الـ WAN واكبر من LAN و تتصل بالانترنت و تحتاج الي شركة اتصالات ISP .. مساحتها اقل من $100km$ زي مدينة مثلً .. والسرعة بتوصل داخل الـ MAN الى $40Gbps$.

3. **Wide Area Network - WAN**: هي عبارة عن مجموعة شبكات الـ LANs مرتبطة مع بعضها عادتاً تكون بين الدول .. و بتتصل بالانترنت و تحتاج إلى شركة اتصالات Internet Service Provider ISP .. و مساحتها اكتر من $100KM$..

4. **Storage Area Network - SAN**: عبارة عن Data Center فيها مجموعة كبيرة من الـ Disk و بتتصل بالانترنت و بتتحتاج إلى شركات اتصالات مثل Switch و بيتوصل به Ram, Processor & NIC و بيتوصل به Hard Disk Arrays . Datacenters عشان يدعم السرعات العالية اللي بنحتاجها في الـ Optical Interface

1.6. Signal Transmission Modes

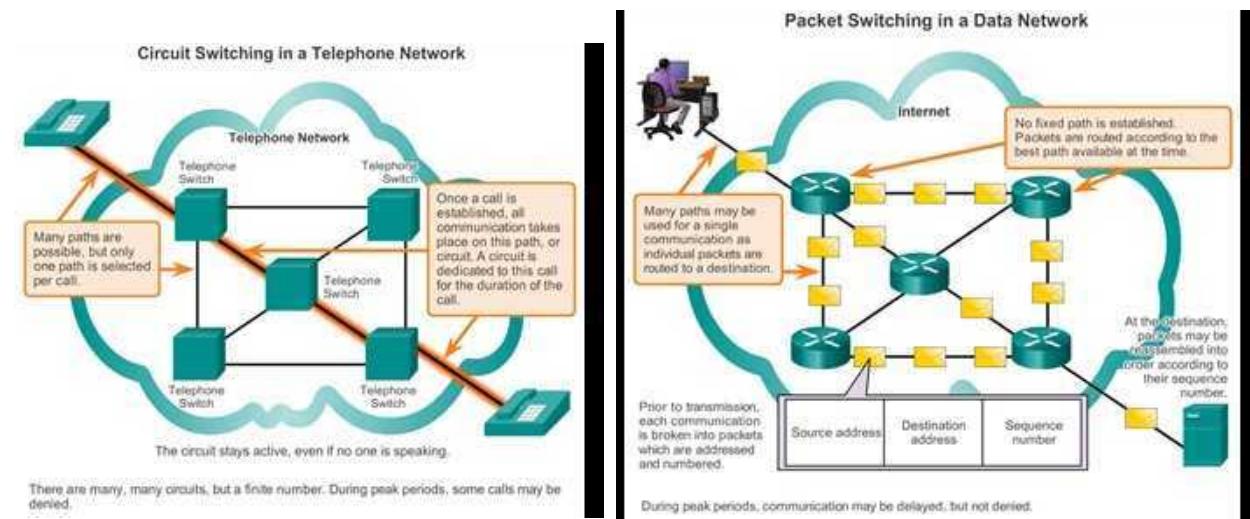
- **Simplex**: Transmission in one direction only
يرسل أو يستقبل في اتجاه واحد فقط زي الراديو او رسیفر القنوات
- **Half Duplex**: Transmission in the two direction but not at the same time
يرسل و يستقبل لكن مش ف نفس الوقت زي اللاسلكي
- **Full Duplex**: Transmission in the two direction at the same time
يرسل و يستقبل ف نفس الوقت زي التليفون

1.7. Network Reliability

الـ Network Reliability هي قدرة الشبكة على أداء وظائفها باستمرار من غير مشاكل واعطال .. ودي بعض المصطلحات الخاصة بالـ Network Reliability :

Fault Tolerance ♦

سماحية الاعطال هي الحد من تأثير سقوط الخدمة في الشبكة .. عن طريق الحد من عدد الأجهزة المتأثرة بالاعطال ودعمها بمسارات متعددة لتحقيق مبدأ الـ Redundancy، وده بيتم في الـ Packet Switched Networks اللي بتعتمد على تقسيم البيانات وترقيمهها مع إمكانية نقلها من خلال أكثر من مسار .. أما في الـ Circuit Switched Networks بيتم اعتماد مسار واحد لكل Connection.



Scalability ♦

لازم أراعي في تصميم الشبكة انها ممكن تحتاج توسيع بسرعة لسهولة دعم المستخدمين والتطبيقات الجديدة من غير ما تأثر على آداء الخدمات للمستخدمين الحاليين.

Quality of Service - QoS ♦

الـ QoS هي التقنية المستخدمة لضمان وصول Traffic معين له الاولوية بطريقة سلية و موثوقة لجميع المستخدمين في الشبكة (زي خدمة الصوت مثلاً).

Security ♦

- على مستوى الـ **Network infrastructure**: تأمين أجهزة الشبكات وتأمين الوصول لها.
- على مستوى الـ **Information Security**: تأمين الوصول للبيانات.

ولازم اراعي تحقيق مبدأ الـ CIA في تأمين الشبكات:

- الحفاظ على سرية البيانات من خلال السماح للمستخدمين الموثوقين فقط من الوصول لها.
- ضمان عدم تغيير البيانات أثناء الإرسال.
- ضمان الوصول للبيانات والـ **Servers** دائمًا أو في أوقات العمل المحددة.

2. Network Model

بنستخدم الـ Network Model عشان نخلي كل الأجهزة الموجودة في الشبكة تفهم بعضها مع اختلاف نظم التشغيل فيها عن طريق استخدام نماذج ذات طبقات بيتم من خلالها تقسيم العمليات اللي بتتم في الشبكة عشان يتم إدارتها بشكل أكثر كفاءة وفهم منه الشبكة بتشتغل ازاي بشكل واضح.

ليه بنستخدم Layered Network Model :

- ❖ بيساعدنا في تصميم البروتوكولات لأن كل بروتوكول محدد بطبيعة معينة فيها معلومات معينة بتخليلها تعرف تتعامل مع الطبقة اللي أعلى منها و اللي أسفلا منها.
- ❖ تعزيز المنافسة لأن المنتجات من Vendors مختلفين ممكن يشتغلوا مع بعض.
- ❖ تقديم لغة مشتركة لوصف وظائف وقدرات الشبكات..

2.1. Standards Organizations

الـ Standards Organizations هي المؤسسات المسؤولة عن وضع المعايير القياسية (Standards) للشبكات من حيث الإنشاء والتشغيل عشان تضمن إن الأجهزة والبروتوكولات المختلفة تشتلل مع بعضها بدون مشاكل. سواء كانت الأجهزة دي من مصنعين (Vendors) مختلفين زي نوكيا وهواوي او مشغلين (Operators) مختلفين زي فودافون واتصالات، لأنهم بيمشو بمعايير وسياسات موحدة .. ودا بيشرح:

- ❖ التوافقية Interoperability
- ❖ المنافسة Competition
- ❖ الابتكار Innovation

المؤسسات اللي بتحط المعايير القياسية (Standard) بتتميز بـ:

- ❖ الحياد مع موردين الأجهزة Vendor-Neutral
- ❖ مؤسسات غير ربحية Non-Profit Organizations
- ❖ Established To Develop And Promote The Concept Of Open Standards
- ❖ المعايير تكون مفتوحة ضد الاحتكار و غير مقتصرة على مورد واحد Single Vendor

في نوعين من المؤسسات الخاصة بعمل الـ **Standard** الخاصة بالشبكات وهي:

• Internet Standards: و هي مؤسسة تتكون من مجموعة مؤسسات مختصة بكل ما له علاقة

بتشغيل الانترنت Public WAN من مستوى 3 Layer إلى مستوى 7 .. واهمهم:

◦ Internet Society - **ISOC**: مؤسسة مختصة بكل ما يخص إنشاء وتشغيل وتطوير الشبكة

العنكبوتية العالمية (الانترنت) من حيث الأجهزة وبرتوكولات تشغيلها.

◦ Internet Corporation for Assigned Names and Numbers - **ICANN**: مؤسسة

مختصة في توزيع عناوين IP وأسماء النطاقات لضمان تنظيم الإنترت.

• Electronic and Communications Standards: عبارة عن مجموعة مؤسسات مختصة بكل ما له

علاقة بتشغيل الشبكات سواء كانت LAN أو WAN.

◦ Institute of Electrical and Electronics Engineers - **IEEE**: هي المسؤولة عن

معايير زي IEEE 802.11 و Ethernet < IEEE 802.3 و Wi-Fi.

◦ International Telecommunication Union - **ITU**: منظمة دولية بتحدد معايير

الاتصالات السلكية واللاسلكية على المستوى العالمي.

◦ International Organization for Standardization - **ISO**: يتغطي معايير زي الـ

.Model

2.2. OSI Model

الـ OSI Model اختصار لـ Network Model وهو أول Open Systems Interconnection (OSI) ي العمل بشكل قياسي عشان يوفر معيار موحد للتواصل بين الانظمة المختلفة (بدل ما كل شركة تستخدمو اسفلها (Model خاص بيها) وتم إنشاؤه وتطويره بواسطة مجموعة مؤسسات زي الـ ISO والـ ITU).

الـ OSI عبارة عن Suit of Protocols ومكون من 7 طبقات .. كل طبقة بتنفذ مجموعة من الوظائف على البيانات باستخدام مجموعة من البروتوكولات.

نبذة عن كل Layer

Application layer (Layer 7)	مسؤولة عن تعريف نوع الـ Service .. سواء Download او Browsing
Presentation layer (L 6)	مسؤولة عن تحديد شكل البيانات (Format) والامتداد بتاعها .. صوت، صورة، فيديو .. الـ Compression والـ Decompression
Session layer (L 5)	مسؤولة عن إدارة جلسة تبادل البيانات (Session) بين الجهازين زي عمل Logical Sessions وتحديد عدد الـ Session للـ Create و termination
Transport layer (L 4)	بيتم من خلالها عمل تقسيم و نقل و تجميع للبيانات .. وبتحدد كمية البيانات اللي هيتم إرسالها للـ Destination من خلال تحديد حجم الـ Buffer والعملية دي اسمها Windowing او Buffering ويتم بالتفاوض مع الـ Dest
Network layer (L 3)	مسؤولة عن نقل وقطع الطلب لقطع (Segments) أقدر انقلها بين أجهزة الشبكة .. ومسؤولة أيضا عن توجيه القطع دي من المرسل للمستقبل ويكون من شبكة شبكة

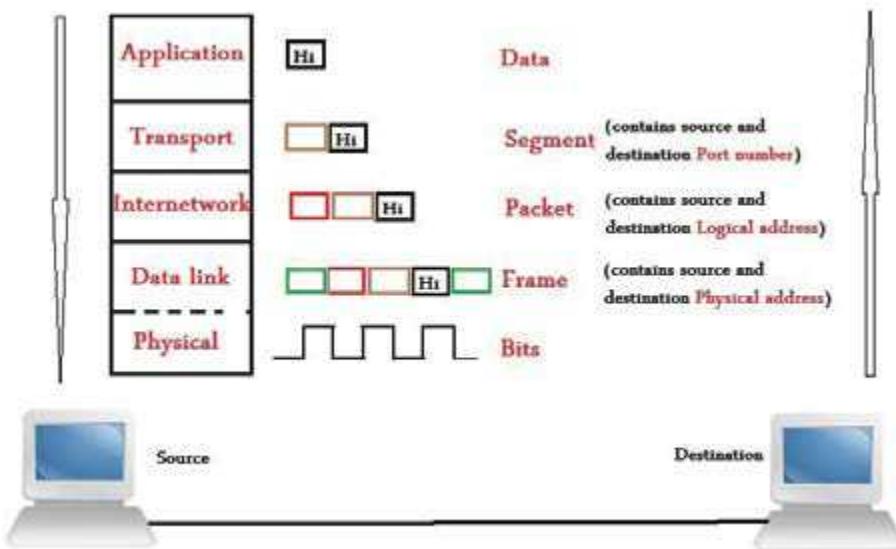
	<p>مسؤولة عن نقل (Switching) القطع من جهاز لجهاز في نفس الشبكة ك خطوه قبل نقلها لشبكة اخرى</p> <p>الـ Sublayer بتنقسم لاثنين Data Link Layer</p> <ul style="list-style-type: none"> • الـ MAC Layer <p>بتضييف الـ MAC في الـ Frame .. ومسؤولة عن ادارة الاتصال زي الـ CSMA/ CA والـ CSMA/ CD</p> <ul style="list-style-type: none"> • الـ LLC - Logical Link Layer <p>من وظائفها .. Multiplexing يعني بتضييف الـ Header المناسب على حسب البروتوكول المستخدم في الـ Network Layer .. و الـ Control</p>
Physical layer (L 1)	<p>مسؤولة عن عمل (Encoding, Signaling) يعني تحويل البيانات من شكل الـ ASCII مثل الى Zeros and Ones او العكس .. ومتقسمة ل حاجتين</p> <p>AMC and LLC - Logical Link Control</p>

لو قلنا ان في جهاز شغال في Layer معينة، مثلا الـ Router عبارة عن Layer3 Device عبارة عن **Layer3 Device** دا معناه ان الجهاز دا بيفهم 3 Layer ويقدر يتعامل معها هي وكل الـ Layers اللي تحتها .. مثلا الـ Switch عبارة عن L2 Physical Layer والـ Data Link Layer .. يبقى بيفهم الـ Device

2.3. Data Encapsulation Process

لو عندنا جهازين او اتنين كمبيوتر - وال Destination Source و Source عايز بيعت بيانات لله البيانات قبل ما تتبعت بتعدى على اكتر من Layer « وكل Layer بتفلغها وتضيف ال Header الخاص بيها وتبعتها لل Layer اللي تحتها .. والعملية دي اسمها .Encapsulation

البيانات بتوصل لل Destination ويبدأ يعمل عكس عملية ال Encapsulation واللي اسمها عشان يوصل للبيانات.



في كل Layer بيبقى البيانات لها اسم مختلف زي ما واضح في الصورة

- "Data" in Application Layer
- "Segment" in Transport Layer for TCP header and called "datagram" if it with UDP Header
- "Packet" in Network Layer
- "Frame" in Data Link Layer
- "Bits" in the Physical Layer

2.4. OSI Layers

2.4.1. Application layer

هي الطبقة رقم 7 في الـ OSI أو رقم 5 في الـ TCP/IP .. وظيفتها تحديد البروتوكول اللي هييفذ الطلب اللي أنت عاوزه وهي برضو حلقة الوصل بين المتصفح وبقى طبقات الـ TCP/IP.

طب بتحدد البروتوكول إزاي ؟

الأول كلمة بروتوكول معناها مجموعة خطوات ثابته لتنفيذ مهمه محدده، فمثلا الـ Download معناه اني هاخد نسخة من ملف ما موجودة على جهاز واطحها على جهازي عن طريق الشبكة .. عشان دا يتم لازم استخدم بروتوكول اسمه .FTP (File Transfer Protocol)

الـ FTP هو البروتوكول المسؤول عن الـ Download وهو اللي بيحدد بالضبط ازاي الموضوع هيتم، ف دور طبقة الـ Application إنها تشو夫 أنت عاوز خدمة إيه وتبعث طلبك دا للبروتوكول المسؤول عنه وبرضو من أشهر بروتوكولاتها >> الـ HTTP دا المسؤول عن التصفح (Browsing).

والفرق بينه وبين الـ Download إن مش شرط تاخد عندك نسخه من الملف على جهازك بشكل دائم عشان تقدر تشووفه، بل يكفي إن يبقى بينك وبين الجهاز الثاني اتصال عن طريق الشبكة وتقدر تشووف كل الملفات بتاعتو عن بعد.

2.4.2. Presentation Layer:

Provides for common representation of the data transferred between application layer services.

هي الطبقة اللي بتحدد نوع البيانات وبتحدد امتدادها (Extension) المناسب، ونوع الترميز (Encoding) المناسب

Text → ASCII ... Picture → JPG, GIF ... Video → Avi, MP4 ... Voice → MP3

2.4.3. Session Layer

يتم من خلالها إدارة تبادل البيانات .. وتحقق من متطلبات الـ Session عشان تدي أمر للـ Transport لعمل Session Termination او Session Establishment لغرض إيقاف ارسال اي ميل بدون عنوان ... فيطلع رسالة بتقول اننا مش حاطين عنوان ... ونفس الحال في عملية معالجة المشاكل وإنهاء الجلسة بعد التأكيد ان الايميل وصل مثلا.

2.4.4. Transport layer

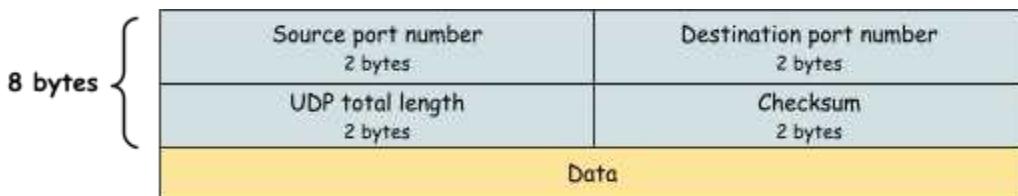
هي الطبقة الرابعة في الـ OSI Model دورها يتمثل في تحديد نوع النقل (TCP/UDP) وحجم الـ Packet ويعمل تقسيم (Segmentation) وتجميع للبيانات. وهي اللي بتحكم فعليا في السيشن من عمل Control و Termination و Establishment.

الـ Transport Layer بتضيف على البيانات Destination Port و Source Port .. ودا العنوان اللي الاجهزه بتعرف منه نوع الخدمة المطلوبة. فمثلا بروتوكول الـ HTTP، الجهاز بيفهمه برقم 80 .Port Number = 80

طريقة العمل؟

بعد ما نوع الخدمة اتعرفت من الـ Application Layer .. بعدها البيانات بتنزل لطبقة الـ Transport

- لو نوع الخدمة بيستخدم بروتوكول UDP زي الـ DNS مثلا >> الـ Transport Layer بتضيف Network Layer على البيانات وتنزلها على الـ Header



الـ UDP Header بيكون من الـ Destination Port والـ Source Port وحجم الـ UDP Header وحجم الـ Header

بالبيانات اللي معاه، وChecksum بتعمل تحقق بسيط لو في Corruption في الرسالة.

نلاحظ ان الـ UDP Header حجمه 8 بايت فقط .. يعني خفيف ومش بيوفر اي طريقة لاسترجاع البيانات ولا التحقق منها .. عشان كدا بيستخدم دائما مع الـ Live Streaming عشان يبقى نقل البيانات سلس ومش بيتأخر.

- اما لو نوعها TCP « بيحصل للبيانات Sequencing و Segmentation .. يعني بتتقسم ل قطع او تقدر الأجهزة تبعتها بعض في الشبكة، وكل Segment بتاخد Sequence Number segments رقم تسلسلي من 32 بت بيساعد على تتبع القطع دي وهي بتتبع عشان لو حدث خطأ أو خسارة قطعة منهم يبقى رقمها معروف وتقدر تسترجعها.

TCP

بروتوكول TCP بيخلify الاتصال Reliable بمعنى لو الجهاز المستقبل موصلوش قطعة هيطلبها تاني من الجهاز المرسل. اما الا UDP غير من Unreliable بمعنى لو قطعة ضاعت مش هيطلبها تاني.

بروتوكول TCP بيوفر Flow Control .. يعني لو الجهاز اللي بيستقبل أبطأ من اللي بيعت، ممكن يحصل ازدحام او فقد في البيانات .. فبيستخدمو آلية اسمها Sliding Window عشان ينظم حجم النقل.

بروتوكول TCP عبارة عن Connection-based Protocol يعني بيعتمد على آلية لبدء وإنهاء الاتصال، عكس الا UDP اللي بيدخل البيت من غير ما يخطط 😊.

20 - 60 bytes

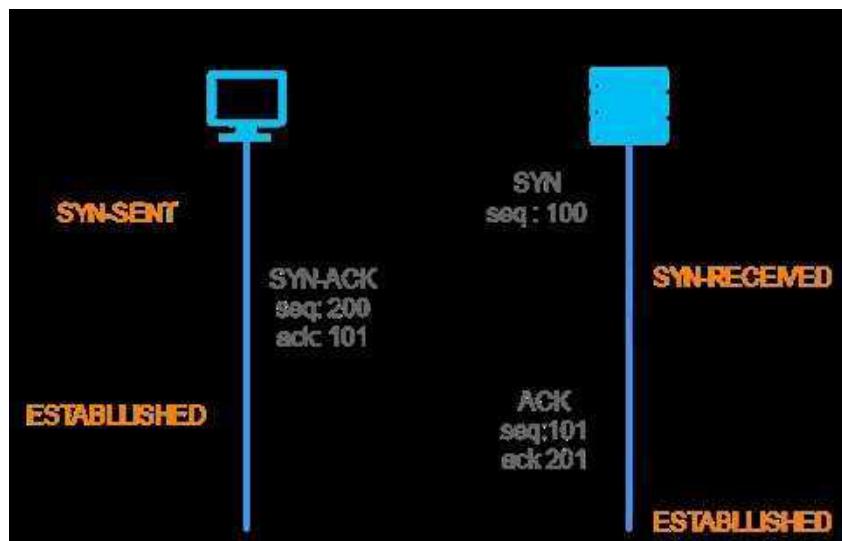
Source port number 2 bytes		Destination port number 2 bytes																			
Sequence number 4 bytes																					
Acknowledge number (ACK) 4 bytes																					
Data offset 4 bits	Reserved 3 bits	Control flags (9 bits)	Window size 2 bytes																		
<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>U</td><td>A</td><td>P</td><td>R</td><td>S</td><td>F</td></tr> <tr> <td>R</td><td>C</td><td>S</td><td>S</td><td>Y</td><td>I</td></tr> <tr> <td>G</td><td>K</td><td>H</td><td>T</td><td>N</td><td>N</td></tr> </table>			U	A	P	R	S	F	R	C	S	S	Y	I	G	K	H	T	N	N	
U	A	P	R	S	F																
R	C	S	S	Y	I																
G	K	H	T	N	N																
Checksum 2 bytes			Urgent pointer 2 bytes																		
Optional data 0-40 bytes																					

مثال بيوضح العمليات اللي بتحصل في حالة استخدام بروتوكول TCP

- اول حاجة البيانات بتتقسم الى Segments .. وكل Segment بتأخذ Sequence Number
- عشان مستخدم يبدا الاتصال مع Server مثلًا .. بيستخدم آلية three way hand-check عشان عبارة عن ٣ خطوات بيتم فيهم التفاوض ما بين الـ Client والـ Server علشان يبدأ الاتصال

بسكل موثوق، والخطوات دي هي:

الـ Client (الجهاز اللي عايز يبدأ الاتصال) بيعت رسالة فيها Flag اسمه SYN، بيقول للـ Server عايز ابدا اتصال معك. والرسالة دي بيبقى فيها مجموعة من المعلومات منها Seq الـ Window بتاعته .. يعني يقدر يستقبل بيانات حجمها قد ايه. وكمان بيبقى فيها Number عشوائي، نفترض انه بيساوي 100.



الـ SYN-ACK: الـ Server بيرد على الـ Client رسالة فيها SYN Flag و اسمه ACK . يعني بيوافق على الاتصال بـ SYN هو كمان، والرسالة بيبقى فيها كمان حجم الـ Window بتاعته.

الـ ACK = 101 معناها إنه استلم SYN بتاع الـ Client (اللي كان رقمه 100) ويقول

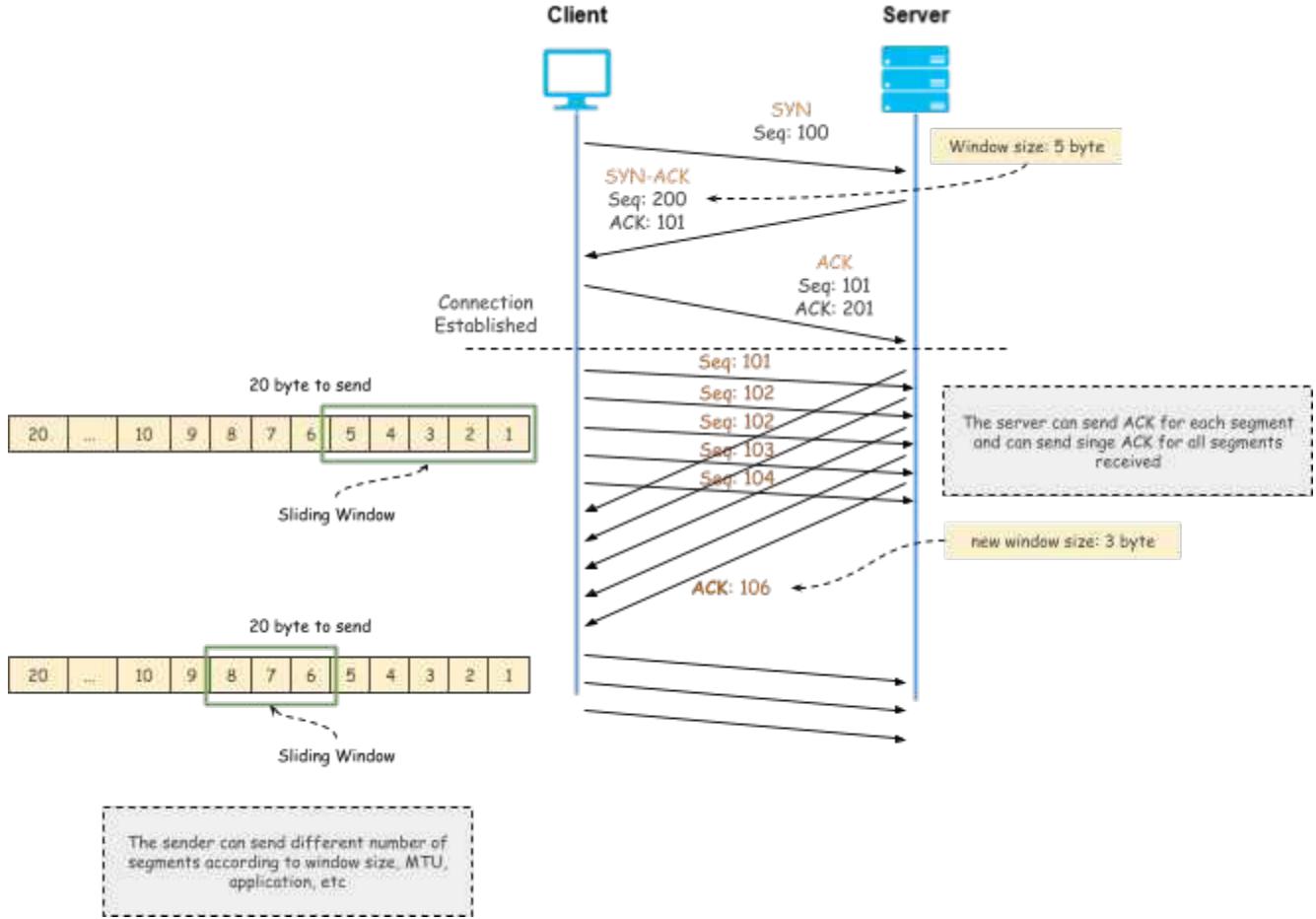
له "أنا مستني منك تبدأ من 101."

الـ Sequence Number = 200 ○ ده الرقم العشوائي اللي اختاره الـ Server لنفسه

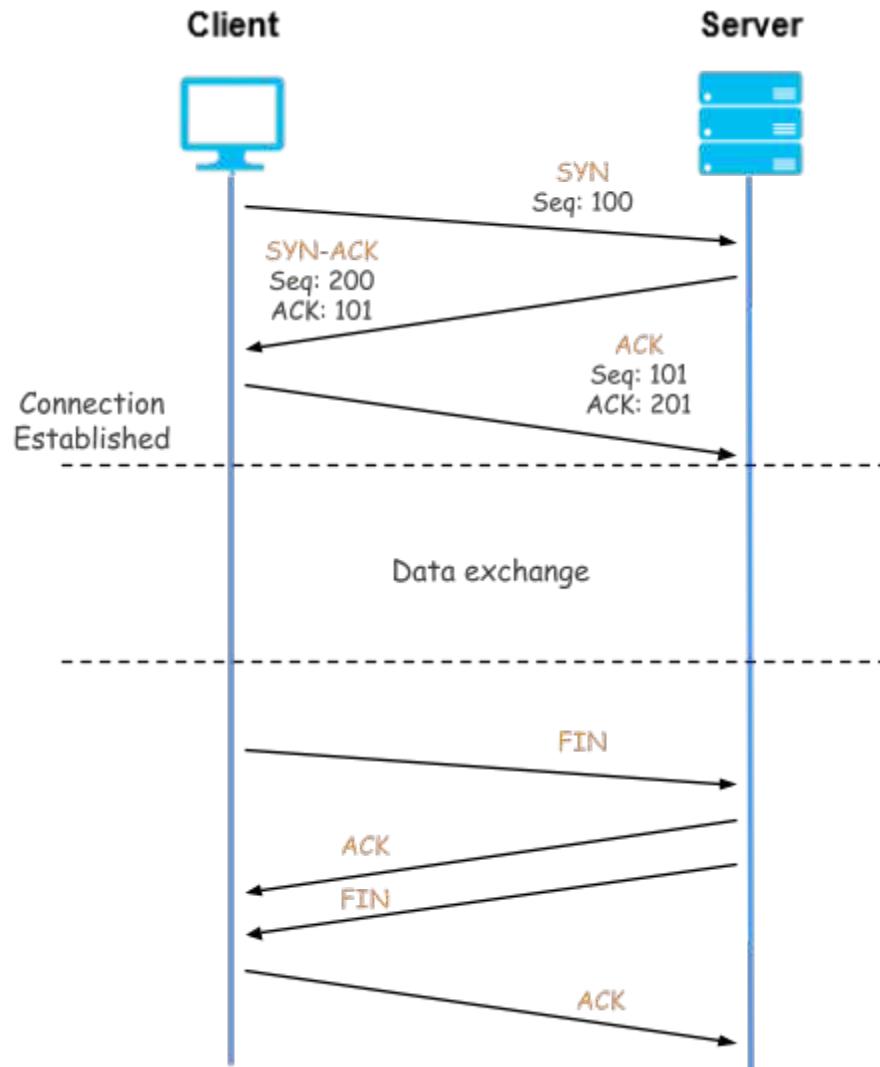
.3 Client الـ ACK بيرد برسالة فيها ACK فقط، للتأكد على استلام الـ SYN-ACK من الـ Server. ومعناها إنه استلم SYN بتاع الـ Server اللي كان رقمه 200، ف بيقوله "أنا مستني منك تبدأ من 201".

- كده الاتصال بقى جاهز، وكل طرف عرف حجم الـ Buffer عند الثاني (Window Size). ويقدر بيتدى بيعت البيانات بطريقة منظمة وآمنة.
- لو افترضنا انه عايز بيعت 20 بايت .. اول Sequence Number هتاخد Segment = آخر ACK استلمه من المستقبل، اللي هو 101.
 - لو حجم الـ Window size عند الـ Server خمسة بايت مثل المرسل مش هينفع بيعت الـ 20 بايت مرة واحدة، لكن هيبيعت اول Segment حجمها 5 بايت .. والـ Sequence Number بتاعها هيبقى 101
 - هتوصل للـ Server .. ولازم يرد بـ Ack عشان ياكد ان البيانات وصلته .. والـ Ack هتاخد رقم 106 يعني بيقول للـ Client انه جاهز يستلم باقي الـ Bytes اللي بتبدا من رقم 106

مع كل ACK .. الرسالة بيبقى فيها حجم الـ Window عشان الـ Server يعرف الـ Client على حجم البيانات اللي يقدر يستقبلها



- بعد الانتهاء من إرسال البيانات .. لو الـ Client عايز ينهي الاتصال بيستخدم Four way and-check
 - بيعت له Segment فيها FIN Flag. يعني بيقله "انا خلصت ارسال بيانات، وعايز انهي الاتصال." ويدخل في حالة اسمها FIN-WAIT-1
 - أول ما الـ Server يستلم رسالة الـ FIN، بيرد على الـ Client بـ ACK يعني بيقوله: "تمام استلمت، رسالتك". ويدخل في حالة اسمها: CLOSE-WAIT و الـ Client بيتحول لـ FIN-WAIT-2
 - بعد ما يخلص هو كمان إرسال بياناته، بيعت للـ Client رسالة FIN. بيقوله: "أنا كمان خلصت إرسال". ويدخل في حالة اسمها: LAST-ACK
 - الـ Client بيرد عليه بـ ACK. بعدها يدخل في حالة اسمها: TIME-WAIT وبيستنى شوية عشان يضمن إن الـ Server استلم الـ ACK وما فيه مشكلة في الشبكة.



2.4.5. Network Layer

الطبقة الثالثة في الـ TCP/IP Model .. دورها الأساسي هو الـ Routing يعني التوجيه.

طب هتوجه إيه وبناء على عنوان إيه؟

- بعد ما الطبقة الرابعة (Transport Layer) تخلص شغلها وتقسم البيانات الى Segments .
- يتنزل لطبقة الـ Network عشان تضيف عليها عنوان IP والـ Segment تحول لـ IP والـ Packet .
- اللي هيخللي الـ Routers في الشبكة تعرف الـ Packet دي جاي من اي عنوان ورايحة لاي عنوان.
- كل Router عنده جدول اسمه الـ Routing Table (جدول التوجيه) ودا لازم يكون فيه Routes (طرق) لكل الشبكات الموجودة، سواء الطرق دي مكتوبة بالتفصيل أو باختصار .. المهم ميكونش فيه شبكيه طريقها مش معروف، وبكدا دور الـ Network Layer هي إنها بتحدد العنوان لك مرسل ومستقبل .. وهي المسؤولة عن توجيه الـ Packet بناء على جدول التوجيه، ومن هنا جه اسم الـ Router يعني الموجه.

الـ Routing Table ممكن أضيف فيه بيانات الشبكات بطريقتين:

- إما Static .. يعني يعرف الشبكات والطرق ليها بطريقة دائمة وثابتة والـ Router هيبص عليها قبل ما يوجه أي Packet حتى لو أنا كاتبها غلط.
- أو Dynamic .. يعني الـ Router من نفسه هيستكشف الشبكة ويعلم نفسه الشبكات المتاحة والمسارات المتاحة لها عن طريق Routing Protocol يعني برنامج توجيه.

2.4.6. Data Link Layer

الطبقة الثانية في الـ TCP/IP Model مسؤوليتها الأساسية هي الـ Switching أي التبديل، يعني نقل الـ Transport Layer من جهاز لجهاز آخر في نفس الشبكة، ودا اسمه Hop-to-Hop Delivery، أما الـ Packet بتعميل End-to-End Delivery يعني توصيل البيانات من جهاز لجهاز في شبكات مختلفة.

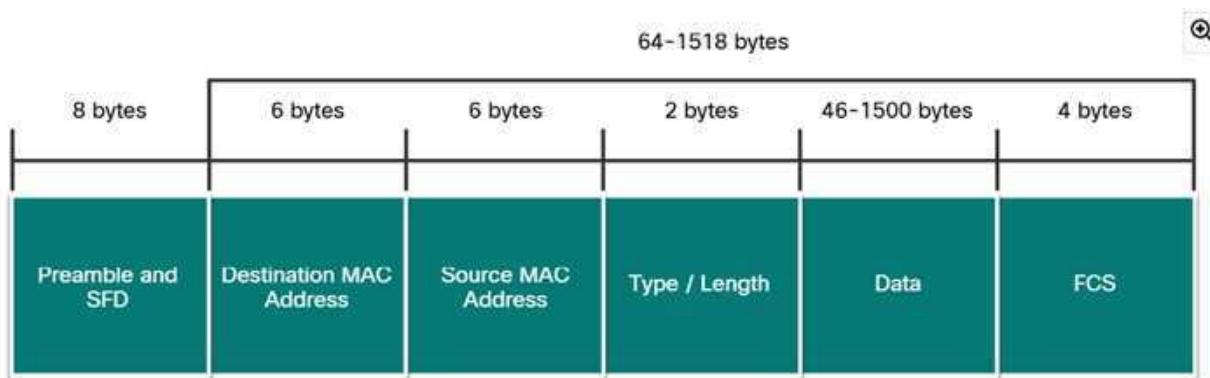
طب النقل من جهاز للثاني بيعتمد على إيه؟

بيعتمد على عنوان اسمه الـ MAC Address، ودا هو العنوان المادي Physical IP لأنه بيبقى موجود على كارت الشبكة وخاص بيها هو فقط.

- بعد ما طبقة الـ Network تخلص شغلها وتنتج Packet.
- طبقة الـ Data Link بتستلمها وتزود عليها MAC الجهاز المرسل و MAC الجهاز المستقبل والFrame تحول له. الجهاز الأساسي اللي بينفذ مهمة الطبقة دي هو السويفتش، وبيعتمد على جدول اسمه MAC-Address Table أو الاسم الأشهر ليه (Switching Table).
- الجدول دا السويفتش بيخزن فيه الـ MAC بتاع كل الأجهزة المتصلة بيها لمدة 5 دقائق (300 ثانية) وعلى أساسه هينقل البيانات من جهاز للثاني.
- السويفتش بيسجل العنوان بتاع الجهاز في حالة واحدة وهي ان الجهاز دا بيعت بيانات تبعي على السويفتش، غير كدا فالجهاز والعنوان بتاعه مجهولين بالنسبة للسويفتش.

2.4.6.1. Ethernet Frame

شكل الـ Ethernet Frame ومكوناته (IEEE 802.3)

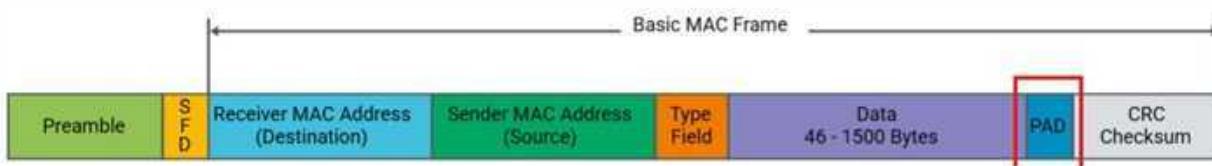


الـ Headers

- اول جزء عبارة عن قيم عشوائية متكررة من الـ (0,1) حجمها 7 بايت، ملحوقة بـ 1 بايت قيمته 10101011 واسمها Start Frame Delimiter - SFD .. ومهما الـ 8 بايت انهم ينبهو الجهاز المستقبل ان في Frame جديد وصل.
- تاني جزء عبارة عن الـ Destination MAC Address وحجمه 6 بايت .. وظيفته ان الجهاز اول ما يستقبل الـ Frame .. يقارن عنوان الـ MAC دا بالعنوان الخاص بي، عشان يعرف هو موجه له ولا لا.
 - والعنوان دا ممكن يكون Unicast يعني خاص بجهاز واحد فقط
 - يعني موجه لكل الاجهزة، زي رسائل الـ ARP، ويبقى قيمته FF:FF:FF:FF:FF:FF Broadcast
 - او .. يعني موجه لمجموعة من الاجهزة .. ويختلف على حسب البروتوكول.
- تالت جزء عبارة عن الـ Source MAC Address الخاص بالجهاز المرسل .. حجمه 6 بايت
- رابع جزء عبارة عن 2 بايت بالـ Hexadecimal يتم استخدامهم لتعريف البروتوكول المستخدم في الـ Network Layer، وهي الـ Upper Layer مثلا 0x800 لـ IPv4، و 0x86DD لـ IPv6، و 0X806 لـ ARP .. في بعض المصادر ممكن تلاقي الـ Field دا اسمه Type او Length او EtherType او ARP

البيانات Data Field

- البيانات الـ Payload اللي معموله Encapsulation جوا Ethernet Header .. ويبقى عبارة عن 3 L3 Packet او بنقول عليه عموما PDU
- حجم الـ Payload يتراوح بين 46 بايت الى 1500 بايت .. وحجم الـ Frame كله لازم ميقلش عن 64 بايت .. ولو قل عن كدا >> يتم اضافة قيم عشوائية في الـ Padding اسمه Filed عشان يصل للـ Minimum



- FCS او Frame Check Sequence وهي عبارة عن Err-Detection Algorithm وظيفتها تتأكد إن الـ Frame محصلش في تلف أثناء انتقاله.
- بتحسب قيمة معينة بناء على البيانات الموجودة في الـ Frame وتضيف القيمة دي في نهاية الـ Frame والتي حجمها 4 بايت، ولما الـ Frame يوصل للجهاز المستقبل بيطبق نفس الـ Algorithm وبيقارن النتيجة بالقيمة اللي وصلت مع الـ Frame .. لو تم اكتشاف خطأ بيعمله Drop.
- احد خوارزميات الـ FCS المشهورة هي CRC او Cyclic Redundancy Check.

ملحوظة:

حجم الـ Frame يتراوح بين 64 بايت الى 1518 بايت .. ولو لأي سبب الـ Frame وصل لجهاز تاني وحجمه أقل من 64 بايت «» يتم تجاهله، يعني بيحصله Drop. وبيطلق عليه في الحالة دي "runt frame" يعني اطار قزم او "collision fragment" مع العلم انه مش بيعمل Collision ولكن بيطلق عليه الاسم دا.

- الـ Preamble مش محسوبة ضمن حجم الـ Frame النهائي .. اللي اخره 1518 بايت.
- أي Frame يقل عن الـ Minimum او يبعدي الـ Max بيحصله Drop .. لكن في اجهزة بتدعم احجام اكبر من كدا على حسب نوع الـ Interface المستخدمة .. مثلاً في في Jumbo Frames ممكن توصل لـ 9000 بايت ولها اسم تاني برضو "baby giant frames".

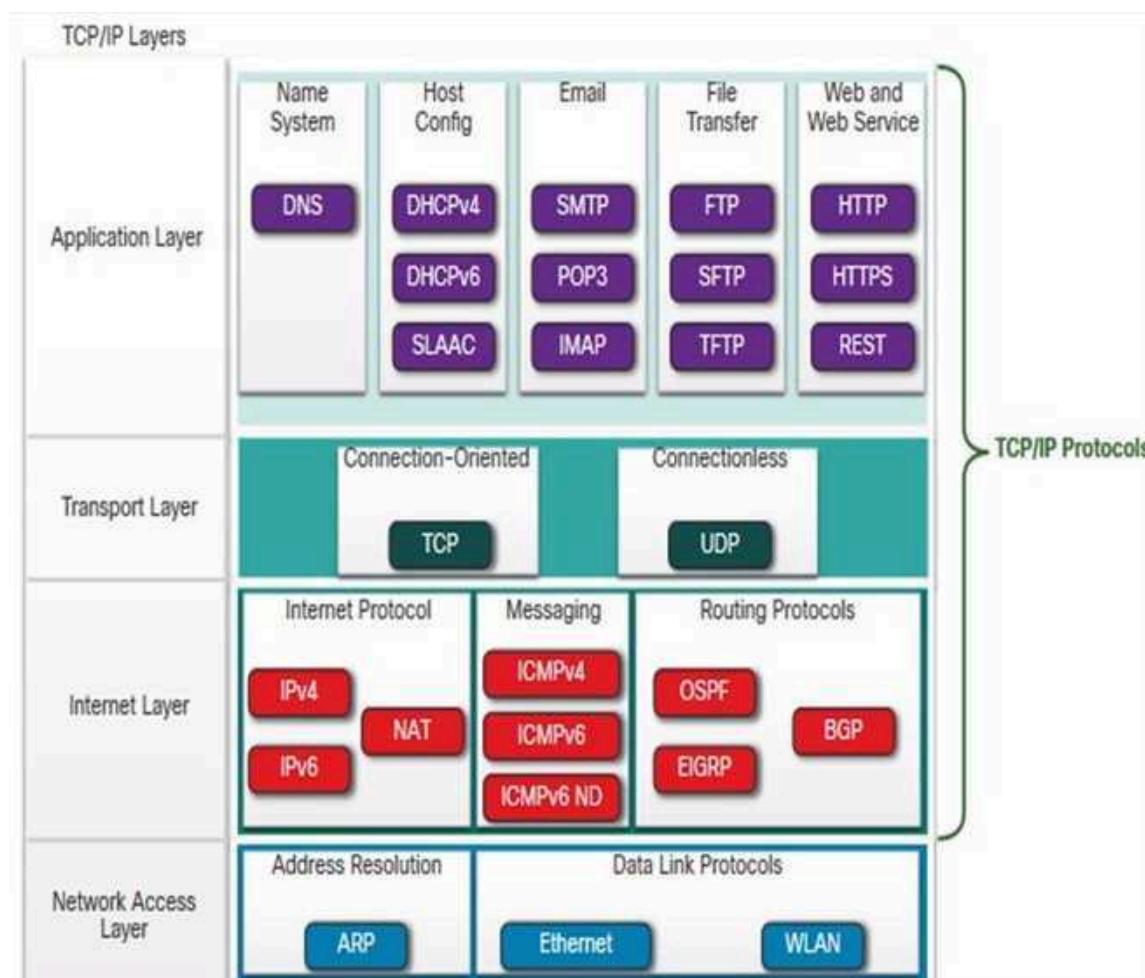
لمعلومات اكتر .. راجع المصدر دا:

<https://pingfu.net/reference/ethernet-ip-tcp-udp-icmp-protocol-header-cheatsheets/>

2.4.7. Physical layer

الطبقة الاولى في الـ TCP/IP Model .. وظيفتها تحويل البيانات اللي بتوصل من الـ Upper Layers على شكل لشكل من أشكال الطاقة يمكن نقلها زي مثلا Electrical Signal تتحرك في كابل أو موجات تتنقل في الهوا أو Optical Pulses.

لو هتنقل كهربا ف انت هتستخدم Media أو أداة توصيل زي كابل UTP ودا المادة الناقلة فيه هي الـ Copper. مميزاته انه سهل التركيب ومش غالى بس مسافته قصيرة وسرعته مش اعلى حاجه، لو حابب مسافات أبعد وسرعات أعلى يبقى تستخدم كابل الفايبر Fiber Optic. ودا المادة الناقلة فيه بتبقى نوع من البلاستك أو الزجاج وبينقل نبضات ضوئية .. لو مش حابب تبقى محکوم بالكابل. ممكن تستخدم الـ Wireless وساعتها البيانات هتحول لموجات تتنقل بين الأجهزة اللي بنقول عليه الـ Wifi.



2.5. Application Layer Services Overview

- DNS اختصار ل Domain Name System: يترجم اسم ال Domain ل IP و العكس
- DHCP اختصار ل Dynamic Host Configuration Protocol: سيرفرات ال DHCP تقوم بتوزيع IP لكل جهاز يقوم في الشبكة و ممكن تعين استخدام اي IP إذا لم يعد له استخدام من جانب الجهاز اللي اخده قبل كده (الجهاز انطفى أو خرج بره الشبكة).
- SLAAC اختصار ل Stateless Address Autoconfiguration: هي طريقة بتسمح للأجهزة بالحصول على IPv6 بدون DHCPv6 server.
- Remote Login: Remote Login اختصار ل Telnet: وده تطبيق مهم جدا بيغدو في عمل Tele Network على الأجهزة وعمل Clear-Text Remote Configuration.
- SSH اختصار ل Secure Shell: وده تطوير لل Telnet بيشفر ال Traffic.
- Email: SMTP اختصار ل Simple Mail Transfer Protocol: بيسمح للأجهزة بارسال واستقبال ال Emails.
- POP3 اختصار ل Post Office Protocol version 3: بيسمح للأجهزة بتنزيل الايميلات من Mail Server الموجود على الجهاز (يعمل Cut لـ Mails) و ساعتها Mail Server من الـ Mail Application.
- IMAP اختصار ل Internet Message Access Protocol: بيسمح للأجهزة بتنزيل الايميلات (يأخذ Copy) من الـ Mail Server للـ Mail Application الموجود على الجهاز الى جانب ان الايميل هيفضل موجود على الـ Mail Server.

File Transfer •

ـ اختصار ل **FTP**: بروتوكول بيسمح للمستخدم انه يبعث ويستقبل

ـ ملفات من والى جهازه ويتميز البروتوكول بأنه **connection-oriented** بس بطء شوية.

ـ اختصار ل **SFTP**: زي بروتوكول الـ **FTP** بس بيشفر الـ

Traffic.

ـ اختصار ل **TFTP**: ابسط من بروتوكول الـ **FTP** ويستخدم

Connectionless. لنقل الملفات الخفيفة و يكون

Web and Web Service •

ـ اختصار ل **HTTP** بيسمح بتبادل النصوص اللي بتحتوي

ـ على صور و صوت وفيديو عبر شبكة الانترنت، ولكنه **Clear-Text**.

ـ بروتوكول **HTTPS** زي بروتوكول الـ **HTTP** بس مشفر.

2.6. Port Number

كل بروتوكول في الشبكة له Port Number خاص بيـه .. وممكن نعتبره منفذ لخدمة معينة على الـ Server اللي بيقدم مجموعة من الخدمات. فمثلا لو عندنا Web site Server وكمان بيقدم خدمة الـ DHCP .. « عشان اوصل للـ Server هحتاج الـ IP .. وعشان اوصل لخدمة معينة للـ Server هحتاج الـ Port Number الخاص بالخدمة دي.

- الـ Port Number بيتمثل في 16bit .. يعني بيوفـر $2^{16} = 65536$ وينقسم لقسمين:
 - Well Known : 0 → 1023
 - User Defined: 1024 → 65535
- الـ Src والـ Dest بيبقى لهم Source .. ودائما عشان الـ Port Number يوصل لخدمة معينة بيطـلـع من فئة الـ Random Port Number
- المفروض ان قسم الـ Well Known محفوظ للبروتوكولات .. ولكن بسبب ان عدد الخدمات والبروتوكولات كـثير جدا .. فـتم تقسيـم الـ User Defined لـقـسمـين إضافـيين:
 - Registered: 1024 → 49151
 - Dynamic: 49152 → 65535
- قـسم الـ Registered بـقى امتداد للـ Well Known .. وما زـال المستخدم يقدر يطلع من الـ Port Number في الـ User Defined كاملـة
- الـ Destination Port هي المسـؤولة عن اضـافـة الـ Source Port والـ Transport Layer على البيانات
- أي Port مـفتوـح على الـ System مـمـكـن يـؤـدي إلـى اختـراق .. عـشـان كـدا يـفضل إـغـلاق الـ Ports أو الخـدمـات الغـير مستـخدـمة

2.7. Application Layer Services

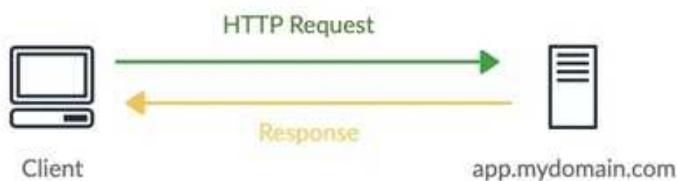
2.7.1. HTTP

بروتوكول HTTP اختصار ل HyperText Transfer Protocol وشغال على Port: 80 TCP

- وظيفة البروتوكول هي فتح صفحات ال Internet
- صفحات ال Web تكون موجودة على Web Server وهو جهاز كمبيوتر عليه Operating System
- .IIS و Apache .. وعليه Web Server بتحوله الى Software Linux او Mac او Windows زى

طريقة عمل البروتوكول:

- اول حاجة ال Client بيكتب رابط الموضع - URL - في ال Browser ويضغط Enter



- ال Server بيعد علىه بـ HTTP Request من Browser وال Server يرجع بـ HTTP Response

- ال Request يتكون من Request Line وBody وHeader



ال Request Line يتكون من:

- ال Request Method اللي بتحدد انتا بتطلب بيانات ولا بتدخل بيانات او غيره.
- ال Request-URL وهو مسار الصفحة المطلوبة.
- HTTP version string الاصدار:

- كل لها وظيفة معينة Methode



◦ مكونات الـ HTTP Response



- اللي يهمنا دلوقتي في الـ Response هو الـ Status Code اللي باللون الأخضر .. ودا بيعرفنا

:Status Codes حالة الصفحة اللي طلبناها .. واشهر الـ

■ 200 يعني الصفحة سليمة وتم استلام Content

■ 404 يعني الصفحة مش موجودة او الموضع واقع

- مشكلة الـ HTTP ان الـ Traffic بيمشي في الشبكة Clear Text يعني مش متشفر .. ودا

عرض للـ Man In The Middle Attack

- عشان كدا عملو بروتوكول الـ HTTPS واللي بيشفر الـ Traffic .. وشغال على بورت 443

TCP

2.7.2. FTP

بروتوكول FTP اختصار لـ File Transfer Protocol وشغال على Port: 20, 21 TCP وهو بروتوكول يستخدم لنقل الملفات وتنظيمها (Download and Upload) .. يعني ممكن تعمل انشاء وحذف لـ Directories على Remote Server.

- بروتوكول الـ FTP بينقل الملفات بشكل غير آمن ClearText .. عشان كدا ممكن نستخدم FTPS او SFTP اللي بينقل الملفات بشكل مشفر عن طريق بروتوكول الـ SSL
- الـ FTP Server عبارة عن كمبيوتر عادي عليه Software يتحوله لـ FTP Server
- بروتوكول الـ FTP بيستخدم بورت 20 لنقل البيانات .. وبورت 21 لنقل الـ Control Commands

2.7.3. SMTP

Simple Mail Transfer Protocol

Port: 25 TCP

الـ SMTP عبارة عن كمبيوتر عليه Software يتحوله الى Mail Server ذي Exchange Mail Server من Sendmail و Postfix و Zebra و IBM و Lotus و Microsoft

- الـ SMTP بوظيفته انه يبعث الرسائل من الـ Client لـ Server فقط
- من البروتوكولات الأخرى اللي بتتوفر خدمة الـ Mail Server
- بروتوكول POP3 ووظيفته انه يحمل الرسائل من الـ Mail Server ويعملها Cut من الـ Server
- بروتوكول IMAP بيعمل للرسائل Sync .. يعني بيحملها ويسيب Copy منها على الـ Server
- بروتوكول ActiveSync بيسقبل الرسائل .. وميزته انه بيعمل للرسائل Push .. بعكس الـ POP3 والـ IMAP اللي بتحتاج انك تحمل الرسائل بنفسك

- POP3: 110 TCP
- POP3S: 995
- IMAP: 143
- IMAPS: 993

2.7.4. Telnet

Port: 23 TCP → Function: Remote Management

- الـ Telnet هو بروتوكول يستخدم لادارة الاجهزة عن بعد من خلال الـ CLI - Command Line Interface .
 - دائما الـ Telnet من رووتراط وسوينشات وكمان الـ Servers بتكون في غرفة باردة جدا اسمها الـ Server room او الـ Datacenter .. ولتسهيل إدارة الأجهزة دي من على بعد بنستخدم بروتوكول الـ Telnet
 - مشكلة بروتوكول الـ Telnet انه ينقل الـ Traffic من غير تشفير Clear Text .. عشان كدا عملي بروتوكول الـ SSH .. وهو اختصار لـ Secure Shell وشغال على بورت 22
- من بروتوكولات الـ Remote Management المشهورة هو بروتوكول الـ RDP او Remote Desktop او من بروتوكولات الـ Graphical User Interface Windows وشغال على بورت TCP 3389

2.7.5. DHCP

Dynamic Host Configuration Protocol | Port: 67 for Servers, 68 for Clients → UDP

- الـ DHCP هو طريقة للحصول على IP للأجهزة الموجودة في الشبكة بشكل Dynamic .
 - في أربع طرق للحصول على IP :
 - Manual
 - Dynamic (DHCP)
 - Alternet
 - APIPA

الجهاز بيتمشي على الأربع طرق دول بالترتيب .. Alternet ثم Dynamic ثم Manual ثم APIPA

1. Manual

هي الطريقة اليدوية لتغيير الـ IP <> فمثلا في Windows ممكن تغير الـ IP من:

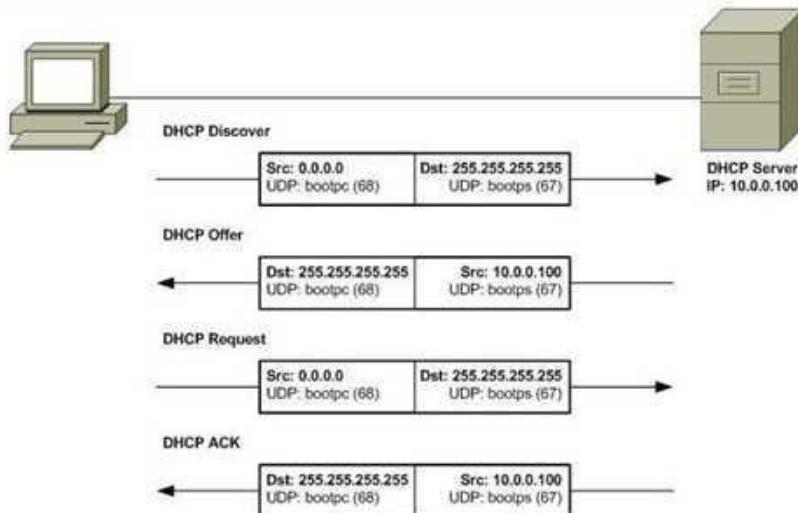
- Control Panel / Network and Internet / Network and Sharing Center
- Change Adapter Settings
- Choose your Network adapter then click properties
- Double Click on Internet Version 4 (TCP/IPv4)
- Chose "Use the following IP" then write your IP

2. Dynamic (DHCP)

الـ DHCP Server هو جهاز كمبيوتر عليه Software يحوله الى DHCP Server .. وبنعمل عليه Pools فيها مجموعة من الـ IPs عشان تتوزع على الـ Clients

طريقة عمل بروتوكول DHCP

- اول ما الجهاز يقوم او اول ما تشغله WiFi Adapter مثلاً هي بعثة Producast Message . DHCP Server من أي Offer وبعد حين هيستنى DHCP Discover
 - لو في اكتر من DHCP Server في الشبكة .. كل واحد بيفرد بـ اسمها Message . DHCP Offer وفيها DNS, Gateway, Subnet Mask و Options + مجموعة من الـ IP Address المقترن زى الـ IP Address وغيرها.
 - الجهاز بيقبل اول Offer وصله .. ويفرج على الـ DHCP Server بـ Request . ويتكون DHCP Request .. يعني بتوصل لكل الـ Servers ودا بيخللي اي Server يستقبل الرسالة مش بيحصل على IP اللي كان مقدمه في الـ Offer .
 - وبعد حين الـ DHCP Server اللي قدم الـ Offer بيفرج على الـ Client . DHCP Acknowledge . وبعد حين الـ DHCP Server بيفرج على الـ Client . DHCP ACK
- والعملية دي اسمها DORA وهي اختصار لـ Discover → Offer → Request → Ack .. وكل الرسائل يتم إرسالها في نظام Windows .. أما في CISCO مثلاً، رسالة Offer و Ack يكونون Broadcast . Unicast



الـ Client لما يأخذ الـ IP .. بيكون له فترة اسمها الـ Lease Period .. ويتكون على Windows لمدة 8 أيام وعلى CISCO لمدة يوم واحد By Default .. والـ Client بيجدد الفترة دي من خلال مجموعة من الخطوات:

- بعد 50% من فترة الـ Lease .. الـ Client بيبعت DHCP Server Unicast DHCP Request للـ Client.

عشان يعمل Renew لفترة الـ Lease.

- لو الـ DHCP Server مردش لأي سبب .. سواء الـ Server واقع او في مشكلة في الـ Connection .Producast الـ Client هيعمل Renew تاني بعد 87% من فترة الـ Lease .. بس المرة دي

- لو مفيش رد من اي Server <> الـ Client هيحتفظ بالـ IP لفترة الـ Lease كاملة وبعدين يفقد الـ IP ويأخذ IP من خلال الـ APIPA عشان يقدر يتواصل مع الاجهزة في نفس الشبكة .. بس مشكلة الـ APIPA اه Unrouteable وكمان الترفيك بتاعه بيكون Broadcast، واللي ممكن يسبب مشاكل في الشبكة.

فترة الـ lease بحددها على حسب الـ Business اللي انا شغال فيه .. يعني لو انا في Caffe مثلاً هقلل الفترة دي لمدة ساعتين مثلاً او اكتر شوية عشان العميل مش هيقعد اكتر من كدا في المكان .. وبالتالي اوفر IPs لباقي العملاء .. وفي ملحوظة تانية وهي ان:

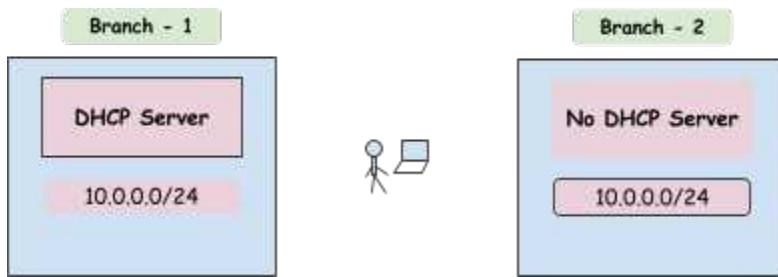
More Requests == More Traffic == More Load

.Lease Period بغير الـ Business وبالتالي على حسب الـ

3. Alternet

الـ **Alternate** يسمح لـ Client انه يأخذ IP لو مش متوفـر Manual IP ولا السيناريو المستخدم لـ **Alternet**

- مثلا عندى شركة صغيرة لها فرعين .. واحد فيه DHCP Server والثانى لا .. وعندى موظف متنقل بين الفرعين (Portable User).
 - بدل ما كل مرة الـ Client يغير الـ IP لما يتنقل بين الفرعين « هحط Alternate IP وبالتالي لو الموظف راح الفرع اللي فيه DHCP Server « هياخد Dynamic IP ولو رجع للفرع اللي من غير .Alternate IP مش هيلاقى DHCP Server فهيروح للطريقة اللي بعدها وهى الـ PC



4. APIPA (Automatic Private IP)

- الاجهزه هتاخد IP في الـ Range :
عن طريق ان كل جهاز هياخد IP في Alternate Range معين تلقائيا.

وعن طريق الـ **Producast** بيتاكدو من الـ IPs عشان ميحصلش **Conflict** .. وبالتالي السيناريو مناسب لـ 5 الى 10 أجهزة مثلا .. لأن لو العدد زاد >> الـ **Broadcast** هتنزيد في الشبكة.

- ال PC لما ياخد APIPA IP بيبعد كل 3 دقائق Producast DHCP Discover عشان يحاول يتواصل مع أى DHCP Server ويأخذ IP.

2.7.6. SMP & NFS

Function:

- **SMP for file sharing in windows**
- **NFS for File sharing in linux**

عشان اعمل **File sharing** بين Windows و Linux لازم استخدم نفس البروتوكول .. سواء باستخدام ال Linux او باستخدام ال **SMP** على Windows او **NFS** على Linux

2.7.7. P2P (Peer to peer)

Function: **File sharing (Download and Upload)**

لما بنستخدم **FTP Server** .. كل ما زاد عدد الـ **Clients** اللي بيحملو من الـ **Server**, كل ما سرعة التحميل قلت بسبب الضغط على الـ **Server** مع زيادة استهلاك الـ **Resources** بتاعته.

اما في الـ **P2P** اللي هو الـ **Torrent Server** :

- يتم تقسيم الملف الى مجموعة من الأجزاء .. واللي يرفع الملف هو اللي بيحدد حجم الأجزاء
- مثلا بيبدأ اول **Client** يحمل اول جزء .. وتاني **Client** بيحمل تاني جزء
- وكل **Client** بيحمل أجزاء الملف من الـ **Server** ومن الـ **Peers** المشاركين معاه في التحميل
- وبالتالي كل ما العدد زاد >> كل ما السرعة زادت

2.7.8. DNS

Domain Name System or Service

Port: 53 / TCP, UDP

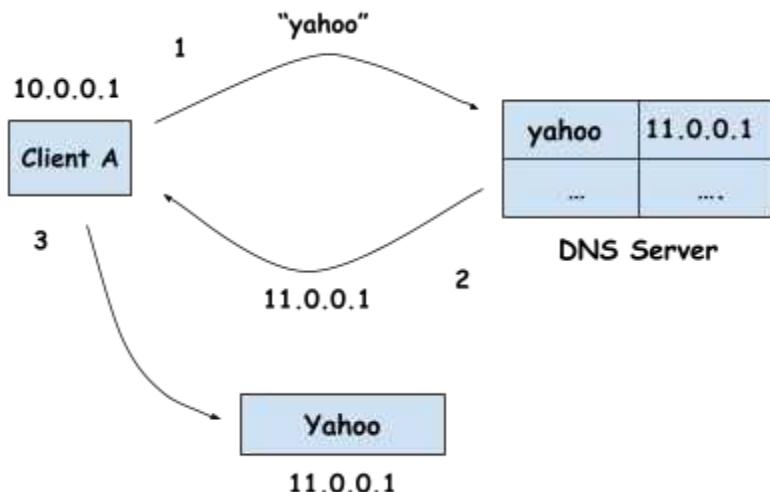
Function: Turn Domain Name into IP address

الـ DNS بيكون عليه Database مكونة من اسم الموقع والـ IP الخاص بي .. وعشان نفهم الـ DNS شغال

ازاي .. هنشرح 3 سيناريوهات

Scenario - 1

- لو Client A عايز يصل لموقع yahoo <> هيكتب في المتصفح www.yahoo.com •
- المتصفح هيبيعت لـ DNS Server بـ IP بـ yahoo عشان يعرف الـ DNS Query •
- الـ DNS هيرد عليه بالـ IP اللي هو 11.0.0.1 •
- بعد كدا الـ Client بقى عارف الـ IP بـ yahoo فهيروح له على طول •



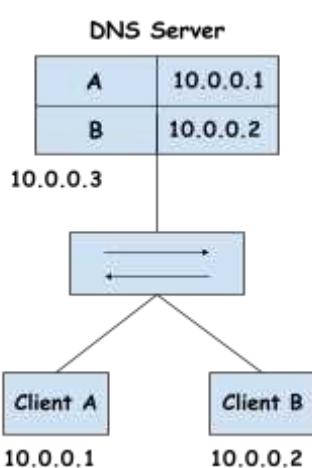
في عدد ضخم جداً من سيرفرات الـ DNS حولين العالم .. على أي أساس Client A راح لـ

8.8.8.8

Scenario - 2

لو عندي DNS واتنين Client متوصلين بسويفتش

- عشان Client A يعمل Sharing Client B مع IP او الاسم



- الـ IP معرض للتغيير .. بس الاسم مش هيتغير، عشان كدا يفضل استخدام الاسم
- لو هتسخدم الاسم » لازم يكون في DNS عشان يربط الاسم بالـ IP
- في الحالة دي ممكن اعمل Sharing عن طريق كتابة الـ IP او اسم الجهاز في قائمة //IP or Name <_ run
- لازم اضبط اعدادات الـ DNS على الأجهزة واخليه 10.0.0.3
- لو الـ IP بتاع الأجهزة اتغير » لازم يتعدل في الـ DNS .. ودا بيتم من خلال ان بيكون في DNS و DHCP Server بين الـ Integration

عشان الـ Client يقدر يصل لموقع معين .. مش منطقى اني اضيف كل المواقع اللي في العالم على الـ DNS

باتاعي .. ولكن في الحالة دي الـ DNS بيتوافق مع DNS Servers تانيين اسمهم الـ Root Hints

الـ DNS عبارة عن 13 Super Computer متوزعين حولين العالم وبيتم تمثيلهم بالـ Dot

- الـ Root Hint Servers يكون متخزن عليهم معلومات عن DNS Servers تانيين اسمهم TLD ..

وفي 3 أنواع من الـ Top Level Domain Servers

edu. و org. و net. و com. زي Generic Top Level Domain <_ ○

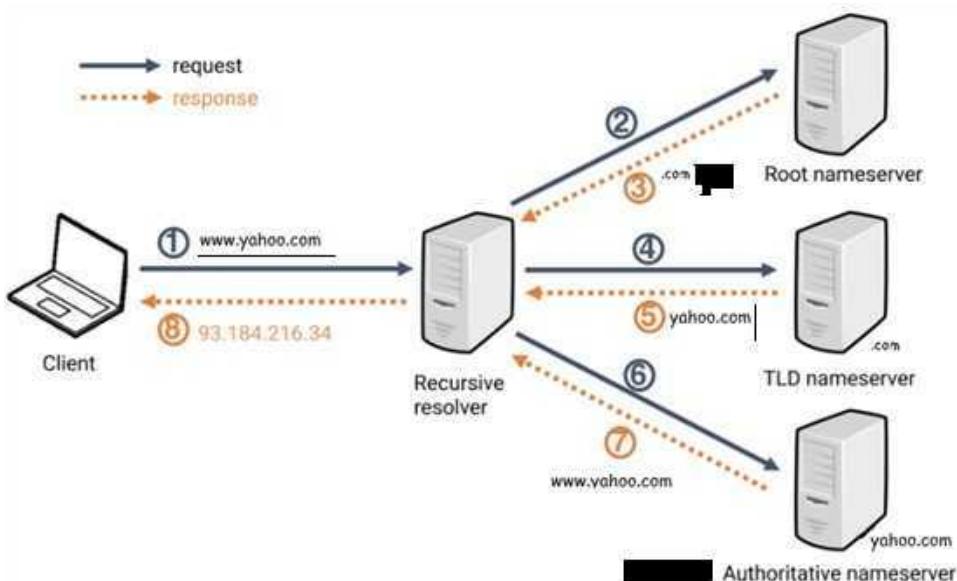
uk. و eg. و us. زي Country Top Level Domain <_ ○

International Top Level Domain <_ ○ وهي سيرفرات خاصة بدول معينة زي الصين ..

وبتكون باللغة المحلية بتاعتهم

لو خدنا مثال على الا GTLD زي com. و yahoo.com و google.com هنلاقي فيه www.yahoo.com و mail.yahoo.com عندها
مثال:

- لو جهاز A عايز يصل لـ "www.yahoo.com" هيسأل الا DNS .
 1. الا DNS هيبحث الأول في قاعدة البيانات بتاعته وبعدين في الا Cash .. لو مفيش هيسأل الا root hint
 2. الا DNS هيرد عليه بالـ IP بتاع الا TLD. COM. < TLD. بـ IP بتاع الا
 3. الا DNS هيروح يسأل .COM على الا IP بتاع "www.yahoo"
 4. الا DNS yahoo يرجع الا IP بتاع الا com.
 5. الا DNS yahoo على الا IP بتاع www .. وهيرد عليه بالـ IP
 6. دلوقتي الا DNS هيرد على الا Client بالـ IP بتاع www.yahoo.com .. والـ Client دلوقتي يقدر يصله



شوية ملحيظ:

- لو الـ DNS طلب IP مش موجود عنده في الـ Database <> بيخزن الـ IP دا في الـ Cash عنده داخل الـ RAM
 - عملة الـ Cash دي بتكون لمدة يوم By Default
 - كمان الـ Client بي العمل TTL لـ IP لمدة معينة بتعتمد على حاجة اسمها الـ TTL
- الـ DNS Forwarder عبارة عن الـ DNS Forwarder
 - يعني ممكن اعمل DNS Forwarder لـ DNS بتعاي بحث لو طلبت IP مش موجود على الـ DNS اللي عندي <> يروح لـ Forwarder
 - والـ Root hint يعتبر بي العمل نفس الوظيفة بس بيبقى Built-in
- الـ DNS Authoritative هو الـ DNS اللي عندو الـ IP في الـ Database بتعاته .. وبيكون لما يجيب المعلومة دي من الـ Cash او من الـ DNS Server تاني
 - الـ Client بيطلب المعلومة من الـ DNS UDP <-
- الـ Super Computers هو الـ BSD اسمه Roothints شغالين بـ Software وهو اختصار لـ Berkeley Software Distribution
- لما الـ Client يطلب معلومة من الـ DNS .. العملية دي اسمها DNS Query .. وفي نوعين منها Recursive Query <-
 - الـ Client بيطلب المعلومة من الـ DNS Server بـ Recursive Query .. وهنـا الـ DNS او بيـقـي اسمـه الـ Recursive DNS Server بيـكون مهمـته يجيـب الـ IP النـهائي بتـاع المـوقـع المـطلـوب

Iterative Query <- ○

ال Iterative Query بتابع ال IP بتاع الموقع النهائي .. لكن
ممكن يكون DNS تاني اقرب .. ودا اللي بيتم بين ال Recursive DNS Server وباقی
سيرفات ال DNS

Some Commands:

- > ipconfig / displaydns
Display all websites opened on the pc .. Most of them are happened in the background
- > nslookup
Display the name of the DNS server and its IP
you can write "nslookup www.google.com" to display information about it

```
> nslookup www.google.com
Server: dns.google
Address: 8.8.8.8

Non-authoritative answer:
Name: www.google.com
Addresses: 2a00:1450:4006:80c::2004
           142.251.37.228
```

3. IPv4

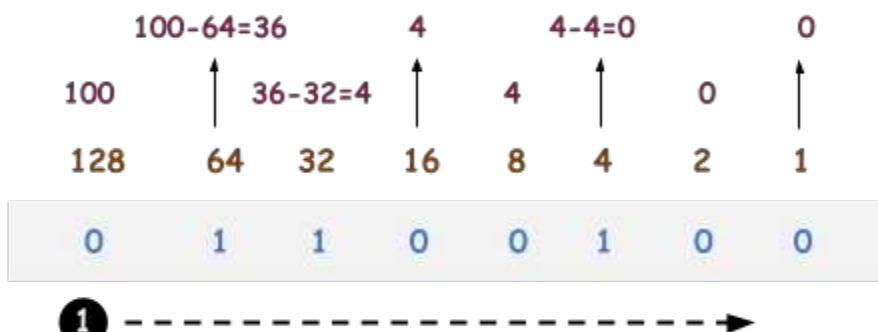
الـ IP عبارة عن اربع ارقام .. كل رقم يسمى Octet يعني bit-8 وبال التالي في 256 احتمال لكل رقم .. يعني ممكن يأخذ قيمة من 0 إلى 255 ومينفعش يبدأ بصفر. قبل ما ندخل على اقسام الـ IP واستخداماته، خلينا ناخذ مراجعة على التحويل من Decimal الى Binary :

مثال لتحويل رقم 100 الى Binary :

- اكتب مضاعفات الرقم 1 في ثمان خانات

128 64 32 16 8 4 2 1

- نبدأ من اليسار .. لو 100 أكبر من الرقم «» نكتب 1 ونطرح من 100 ونأخذ الناتج ونقارنون بالرقم اللي بعدو وهكذا .. لو الرقم اصغر «» نحط صفر



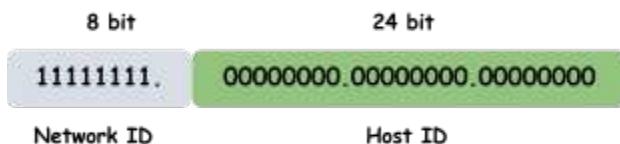
منظمة الـ IANA مسؤولة عن حاجات كتير على مستوى الانترنت وفي تحتها منظمة الـ ICAN واللي تحتها خمس منظمات اخرى مسؤولة عن توزيع الـ IP على مستوى كل قارة والخمس منظمات يطلق عليهم AIR.

3.1. IP Address Classes

Class	Fixed bits	First Address	Last Address	Subnet Mask	Application
A	0	1 10000000	127.255.255.255 01111111	255.0.0.0	Used for a large number of hosts.
B	10	128.0.0.0 10000000	191.255.255.255 10111111	255.255.0.0	Used for medium size networks.
C	110	192.0.0.0 11000000	223.255.255.55 11011111	255.255.255.0	Used for local area networks.
D	1110	224.0.0.0 11100000	239.255.255.255 11101111	no subnet mask	Reserve for multi-tasking
E	1111	240.0.0.0 11110000	255.255.255.255 11111111	no subnet mask	Reserved for research and Development Purposes.

Class A:

Subnet Mask: 255.0.0.0



- Maximum number of hosts = $2^{24} - 2 = 16777214$ host ID

كل Class بيبقى فيها اتنين IP مجازين <> أول واحد هو عنوان الشبكة وآخر واحد يستخدم للBroadcasting

- Maximum number of networks = $2^{(8-1)} - 2 = 126$ Network

بنطرح الـ Fixed bits من عدد الـ Bits بتوع الـ Network

3.2. Private IP Range

- *Class A:*

- 10.0.0.0 to 10.255.255.255
- IP Range: 127.0.0.1 to 127.255.255.255

الـ Range دا بيستخدم في الـ Network testing عن طريق امر ping او كـ Loop-back address .. لكن مينفعش اضيفه لكارت الشبكة العادي ومنش هيقبل اصلا .. ومثلا الـ IP 127.0.0.1 بيستخدم عشان تعمل الـ TCP/IP Test وتعرف هو شغال كويس ولا لا

- *Class B:*

- 172.16.0.0 to 172.31.255.255
- APIPA Range: 169.254.0.0 to 169.254.255.255

Automatic Private IP Addressing (APIPA)

هي Future Microsoft من عشان كل جهاز في شبكة الـ LAN ياخد IP بشكل تلقائي في حالة عدم توفر الـ DHCP Server

- *Class C:*

- 192.168.0.0 to 192.168.255.255
- Class D is reserved for Multicasting while Class E is reserved for experimental purpose for IANA
- 255.255.255.255 is the **general broadcasting IP**

3.3. Ping

هو أمر يستخدم للتحقق من حالة الاتصال بين جهازين عن طريق ارسال مجموعة من الـ Packets وانتظار رد

3.3.1. Ping Command Responses

لو عملت ping على اي IP <> هيظهر أحد النتائج التالية:

- **Reply from [IP Address]**

This indicates that the target is reachable and responding to the ping request.

- **Destination unreachable or General failure**

Destination unreachable in Windows XP or General failure in Windows 7, 8, 10, 11 indicates that the target cannot be reached. Possible reasons include:

- The target device is turned off or disconnected from the network or in another network
- There is a firewall blocking the ping requests.

- **Request timed out in Windows XP:**

This means that the ping request did not receive a response within the expected time frame, suggesting possible network issues or that the target is not reachable.

3.3.2. Ping Command Characteristics

- **Sending Unlimited Packets**

`ping [IP Address] -t`: continuously pings the target IP address until you manually stop it (usually by pressing Ctrl + C).

- **Specifying the Number of Packets:**

`ping -n [number] [IP Address]`: This command allows you to specify the number of ping requests to send. For example, `ping -n 5 [IP Address]` will send 5 ping requests.

- **Ping 127.0.0.1:** If you execute the command `ping 127.0.0.1` and receive a reply, it means that the network interface card (NIC) is functioning and the TCP/IP protocol suite is correctly installed and operational on your computer.

3.4. IPv4 Subnetting

الـ IP مقسم إلى خمسة *Class*, منهم اثنين محفوظين لمنظمة الـ IANA. كل من الثلاثة الباقين يمثل شبكة لوحده، ولو خدت IP من أي *Class* .. كذا باقي الـ IP's هتبقى محفوظة .. فهياحصل فقد كبير للـ IP's عشان كده فكرو في الـ Subnetting .. عن طريق الـ subnetting اقدر أقسم كل *Class* لمجموعة متساوية او غير متساوية من الشبكات.

3.4.1. Fixed Subnetting

في الـ Fixed Subnetting بنقسم الشبكة إلى مجموعة من الشبكات فيها عدد متساوي من الـ Hosts

مثلا لو عايزين نقسم الشبكة 192.168.1.0/24 لمجموعة من الشبكات من 15 IP

- الرمز /24 يسمى Prefix وهو طريقة أخرى لتمثيل الـ Subnet Mask وبيمثل عدد الـ Bits اللي

متساوي 1 من الشمال لليمين

اول حاجة لتقسيم الشبكة لمجموعة متساوية من الشبكات من 15 IP « هنستخدم المعادلة $2^n - 2 = x$ حيث x هي عدد الـ IPs اللي احنا عايزينه .. و n هو عدد الـ Bits المحفوظة للـ Hosts و 2 هو الـ IP المحفوظ للـ Broadcast والـ IP المحجوز لنوان الشبكة

- تاني خطوة هنشوف اقرب رقم للـ n يخلي ناتج المعادلة اكبر من او يساوي 15

$$2^n \geq 15 + 2$$

- اقرب رقم للـ n هو 5 والـ 5 هيدينا 32 IP في الشبكة الواحدة .. وبالتالي آخر Octet في منه

خمسة Bit من اليمين محفوظين لعدد الـ Hosts

- We use the subnet mask with the prefix /24, which is

11111111.11111111.11111111.00000000

- After subnetting, the subnet mask is updated to:

- Subnet mask: 255.255.255.224 (/27)

11111111.11111111.11111111.11100000

- The first subnet will be:

- Network address: 192.168.1.0
- Usable IP range: 192.168.1.1 to 192.168.1.30
- Broadcast address: 192.168.1.31

The zero address is considered, so we increment by 32 each time to get the new block.

Subnet Ranges:

A	B
N1: 192.168.1.0	→ 192.168.1.31
N2: 192.168.1.32	→ 192.168.1.63
N3: 192.168.1.64	→ 192.168.1.95
N4: 192.168.1.96	→ 192.168.1.127
N5: 192.168.1.128	→ 192.168.1.159
N6: 192.168.1.160	→ 192.168.1.191
N7: 192.168.1.192	→ 192.168.1.223
N8: 192.168.1.224	→ 192.168.1.255

A is the network ID, the first address in the subnet.

B is the broadcast address, the last address in the subnet.

Subnetting Steps:

- Determine the required number of hosts:
Find the nearest power of 2 greater than or equal to the number of required hosts plus 2.
- Calculate the new subnet mask:
Subtract the number of host bits from 32 to get the new prefix length.
- Divide the network into subnets:
Use the new subnet mask to calculate the range of each subnet.

3.4.2. Variable Length Subnet Mask (VLSM)

- الـ VLSM هي طريقة أخرى للـ subnetting لتقسيم الشبكة إلى شبكات غير متساوية في عدد الـ IP بافتراض نفس المثال السابق .. لو في عميل عايز IP-15 <> هقسم الشبكة بنفس الطريقة اللي فاتت.
- ولو جالي عميل تاني عايز خمسة IP .. مش منطقي ياخذ شبكة فيها 32 IP وهو عايز 5 بس
 - في طريقة الـ VLSM دائمًا بنحتفظ بأول network لأنها أسهل في التقسيم فيما بعد
 - بنقسم أول شبكة إلى مجموعة من الشبكات فيها على الأقل خمسة IP
 - كدا الشبكة الاولى في المثال السابق اللي هي 192.168.1.0 -> 192.168.1.31 .. هيبيقى الـ 192.168.1.0/27 Prefix
 - هنسخدم نفس الطريقة لتقسيم الشبكة دي إلى مجموعة من الشبكات من خمسة IP .. وه تكون النتيجة كالتالي:
 - subnetmask: 192.168.1.0/29
 - N1: 192.168.1.0 → 192.168.1.7
 - N2: 192.168.1.8 → 192.168.1.15
 - N3: 192.168.1.16 → 192.168.1.23
 - N4: 192.168.1.24 → 192.168.1.31

ملاحظيظ:

اقل شبكة ممكن اعملها بتكون من أربعة IP

واحد للـ Network ID وواحد للـ Products

واحد للـ Router وواحد للـ Gateway

طب ازاي كل بيت فيه اجهزة كتير ولهem IP واحد بس

هنا يأتي دور تقنية الـ NAT + Private IP

بسبب الـ **IPv4** الكبير في الـ **west** اتجهنا الى حلين:

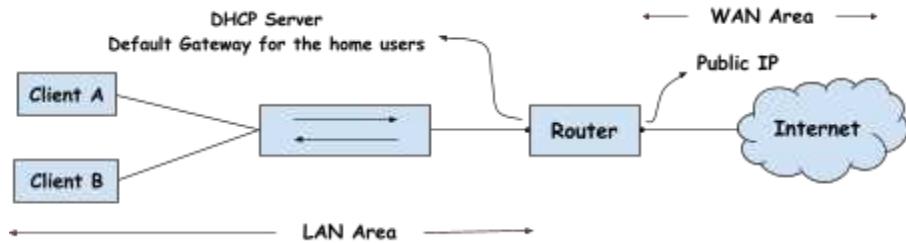
- Long term solution (IPv6)

حل على المدى البعيد وهو إستخدام IPv6 .. وهو على المدى البعيد عشان في شركات كتير بتستخدم برامج قديمة بتشتغل على أنظمة قديمة مش بتدعم IPv6 فمینفعش يعملو تحويل مرة وحدة

- Short term Solution: (Private IP + NAT)

Private IP

منظمة الـ IANA سمحت للعميل باستخدام العدد اللي هو عايزو من الـ IPs بس ميخرجش من الـ **Section 3.2** اللي محدد في Private IPs



• كل Interface على الـ Router يأخذ IP مختلف

• الـ IP Default Gateway يتحط في خانة الـ Router على الأجهزة المتصلة بالـ LAN

لأن هوا دا البوابة اللي هتلعلعني Internet

NAT

• تقنية الـ NAT هي اللي بتخلி كل IP في الـ Private Area يتترجم الى الـ Public IP بتاع الـ Router لما ييجي يطلع Internet .. وبالتالي ممكن نستخدم نفس الـ IPs في اي LAN تانية .. ولو هتواصل مع بعض عن طريق الـ Internet .. الـ Private IP هيتتحول لـ IP اللي بيكون Unique على مستوى العالم

• الـ Home Users العاديين بيكون الـ Public IP بتاعهم مش ثابت .. يعني بيتغير كل فترة (يومين مثلًا) او بيتغير كل ما تعمل Connect لـ Router .. لأن الـ Router هي عمل Connect على الـ ISP

• بتعال الـ ISP عشان ياخد IP .. وفي حالة لو عايز IP ثابت .. لازم تشتري Static IP من الـ ISP

3.4.3. ARP

Address Resolution Protocol

بروتوكول الـ ARP يشتغل في طبقة الـ Data Link Layer في OSI Model ووظيفته إنه يترجم عنوان الـ IP إلى عنوان الـ MAC

- معروف أن كل جهاز على الشبكة بيكون ليه عنوانين:

- العنوان المستخدم للتواصل على مستوى الشبكات IP Address
- والعنوان الفيزيائي لкарت الشبكة MAC Address

طريقة عمل البروتوكول

- عشان تبعث Packet لجهاز تاني في نفس الشبكة تحتاج 4 حاجات أساسية :

src MAC address .. src IP address .. dest MAC address .. dest IP address

- لو مثلاً معاك الـ destination IP /src MAC لأنهم بتوعدك فانت

فاضللك بس الـ destination MAC address .. وبالتالي عشان تعرف الـ MAC المرتبط بال

arp IP Address يتبع الجهاز الثاني .. هتسخدم الـ

- الجهاز بيبيغ رسالة ARP Request وبتكون Broadcast على الشبكة (يعني بتروح لكل الأجهزة اللي في نفس الشبكة)، وبتقول "من صاح الـ IP ده؟".

الـ ARP Request بتكون فيها عنوان الـ IP اللي الجهاز عاوز يعرف الـ MAC الخاص بي.

- الجهاز اللي ليه الـ IP المطلوب هو الوحيد اللي بيرد برسالة ARP Reply، ودي بتكون Unicast (يعني رايحة لجهاز واحد)، وفيها عنوان الـ MAC الخاص بي.

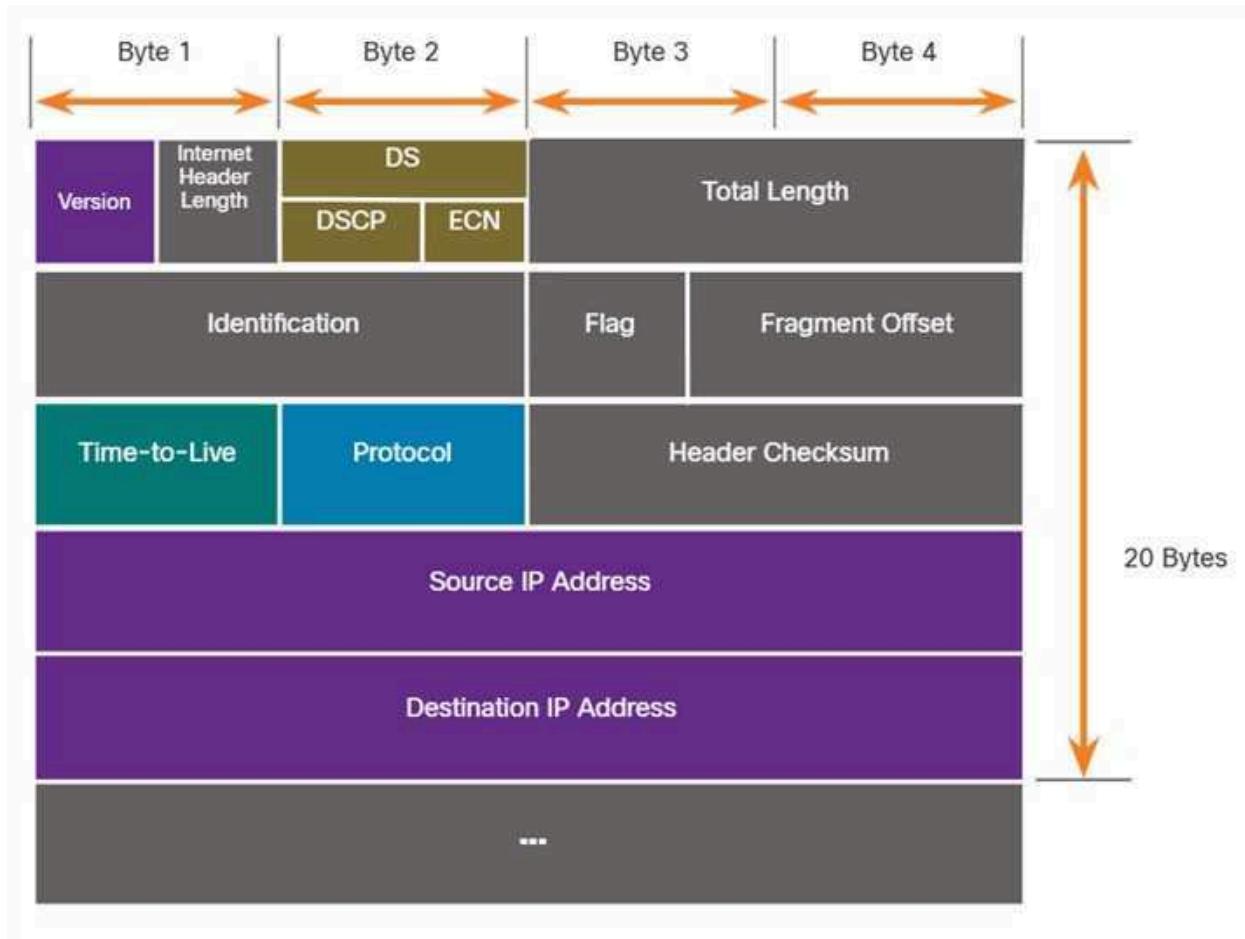
- الجهاز اللي بعث الطلب بيذخن الـ MAC Address اللي استلمه في حاجة اسمها ARP Cache، وده جدول بيحتفظ فيه بالعناوين اللي ترجمها عشان يستخدمها لو احتاج يتواصل مع نفس الجهاز تاني، بدل ما يسأل كل مرة.

- في مشكلة ممكن تظهر هنا .. وهي لو الـ Route اتغير مثلا، وفي الحالة دي هيأخذ نفس الـ IP بس الـ MAC بتاعه هيكون مختلف .. وبما ان الـ MAC بيفضل متخزن في الـ Cache لمدة معينة او لحد ما تعمل اعادة تشغيل للجهاز .. فدا هييعمل مشكلة في الشبكة
 - عشان كدا عملو بروتوكول Gratitius ARP واللي بيخللي الـ Router يقدر بيعرف الـ IP.
 - ARP لكل الأجهزة اللي في الشبكة عشان يعملو Cashe للـ MAC Address الجديد ويمسحو القديم اللي متخزن بنفس الـ IP.
 - ممكن يتم استغلال البروتوكول دا لو في شخص معايا في الشبكة بعث GARP وهو بيستخدم نفس IP الخاص بالـ Router وبالتالي ممكن يطبق الـ Man in the Middle Attack .. ولتفادي المشكلة دي <> ممكن نغير العلاقة بين الـ MAC والـ IP الى Static .. ودا هيوقف بروتوكول الـ GARP.

بعض الأوامر

- أمر arp -a بيستخدم عشان يعرض جدول الـ ARP المخزن على جهازك، وبيحتوي على كل العنوانين اللي ARP ترجمتها مؤخرًا، وده بيشمل:
 - ده عنوان الـ IP اللي الجهاز بتاعك تواصل معاه.
 - ده العنوان الفعلي أو الـ MAC Address الخاص بالجهاز اللي بتتواصل معاه.
 - Type: فيه نوعين .. <> العنوان ده تم تعلمه آليا عن طريق ARP Request dynamic.
 - static: العنوان ده تم تحديده يدوياً أو بيكون عنوان خاص ببروتوكولات معينة زي الـ Broadcast أو الـ Multicast
- أمر arp -d بيستخدم لحذف الـ ARP Cache

3.5. IPv4 Header



Significant fields in the IPv4 header include the following:

- **Version** - Contains a 4-bit binary value set to 0100 that identifies this as an IPv4 packet.
- **Differentiated Services or DiffServ (DS)** - Formerly called the type of service (ToS) field, the DS field is an 8-bit field used to determine the priority of each packet. The six most significant bits of the DiffServ field are the differentiated services code point (DSCP) bits and the last two bits are the explicit congestion notification (ECN) bits.
- **Time to Live (TTL)** - TTL contains an 8-bit binary value that is used to limit the lifetime of a packet. The source device of the IPv4 packet sets the initial TTL value. It is decreased by one each time the packet is processed by a router. If the TTL field decrements to zero, the router discards the packet and sends an Internet Control Message Protocol (ICMP) Time Exceeded message to the source IP address. Because the router decrements the TTL of each packet, the router must also recalculate the Header Checksum.

- **Protocol** - This field is used to identify the next level protocol. This 8-bit binary value indicates the data payload type that the packet is carrying, which enables the network layer to pass the data to the appropriate upper-layer protocol. Common values include ICMP (1), TCP (6), and UDP (17).
- **Header Checksum** - This is used to detect corruption in the IPv4 header.
- **Source IPv4 Address** - This contains a 32-bit binary value that represents the source IPv4 address of the packet. The source IPv4 address is always a unicast address.
- **Destination IPv4 Address** - This contains a 32-bit binary value that represents the destination IPv4 address of the packet. The destination IPv4 address is a unicast, multicast, or broadcast address.

أشهر حاجتين في IPv4 هما إل Destination Address و إل Source Address .. لأنهم يحددوا إل Packet جاية منين و موجهة لمين .. ومش بيتغير أثناء انتقال إل Packet في الشبكة، الا لو مطبق NAT.

إل Fields اللي بتسخدم للتأكد من إل Packet هم:

إل Internet Header Length IHL • حسب إل Options

إل Total Length • بيحدد الطول الكلي لـ Packet (الهيدر + الداتا)، علشان نعرف نهاية إل Packet فين.

إل Header Checksum • بيستخدم علشان يتتأكد إن الهيدر متغيرش أثناء النقل

فيه 3 حقول بيستخدموا لو إل Packet اتقسمت لأجزاء (Fragments) .. مثلاً لما بيحصلها Forwarding من

إلى Medium بـ MTU أقل

إل Identification • وهو رقم موحد لكل إل Fragments اللي جايين من نفس إل Packet الأصلية.

إل Flags • فيها بت، من ضمنهم واحد اسمه MF - More Fragments، لو قيمته 1 بقى لسه فيه أجزاء تانية جاية.

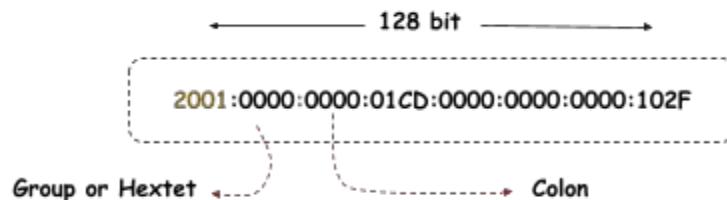
إل Fragment Offset • بيحدد مكان كل جزء داخل إل Packet، علشان نعرف نرتبعهم صح لما يصلوا.

إل Options و Padding نادراً ما بيستخدموا، غالباً مش بتحتاج تعامل معاهم في السيناريوهات العاديّة

4. IPv6

يكتب بنظام الـ HexaDecimal .. يعني اصغر رقم هو صفر، واكبر رقم هو "F" .. وهو عبارة عن 8 مجموعات Groupes، وكل مجموعة عبارة عن اربع ارقام يفصل بينهم ب ":"، وكل رقم عبارة عن اربعة بت

مثال:



4.1. IPv6 Abbreviations

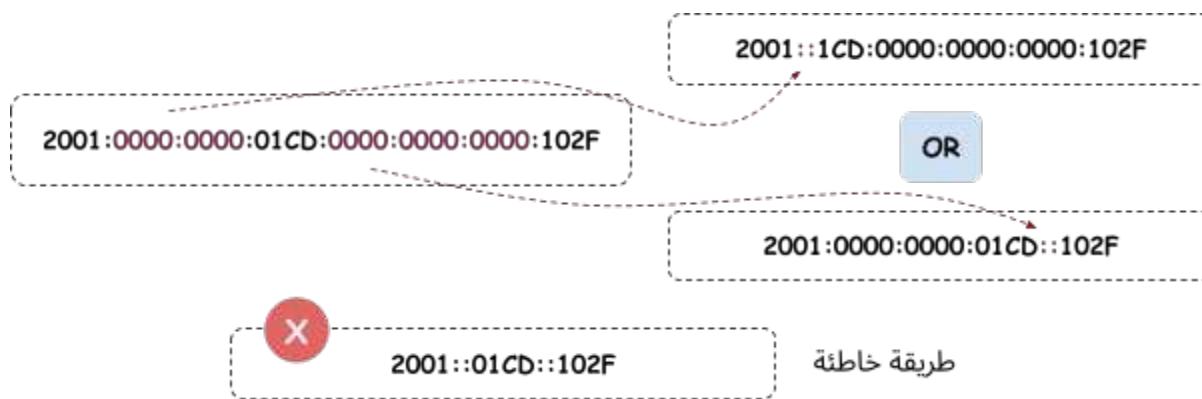
Leading zero compression

أي صفر على الشمال مش بيكتب

2001:0000:0000:01CD:0000:0000:0000:102F 2001:0000:0000:1CD:0000:0000:0000:102F

Zero compression

اختصار أي مجموعة متتالية من الأصفار بصفر واحد



Zero compression with Leading zero compression >> Also called Compressed Form

2001:0000:0000:01CD:0000:0000:0000:102F

2001:0:0:1CD::102F

OR

2001::1CD:0:0:0:102F

4.2. Two methods to get IPv6

- Static

- Manual

You can add or change the IP from Control Panel in windows, or use the command netsh /? and follow the help. In windows xp, you can use command line only

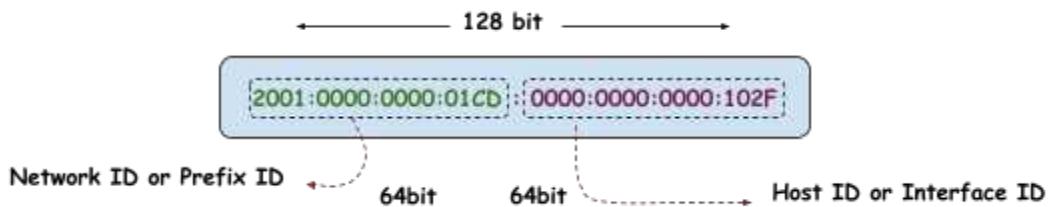
- EUI-64

- Dynamic

- SLAAC - Stateless Add Auto Config
 - Stateless
 - Stateful

4.3. EUI-64 (Link Local Address or LLA)

EUI-64 هي طريقة تستخدم في بروتوكول IPv6 لتكوين عناوين بشكل تلقائي، بناءً على عنوان MAC الخاص بالجهاز.



الـ Prefix ID يكون ثابت على مستوى الـ LAN والـ Interface ID هو اللي بيتغير، وبالتالي كل جهاز ممكن يشتق الـ ID من الـ MAC الخاص بيه.

طريقة الاشتقاء

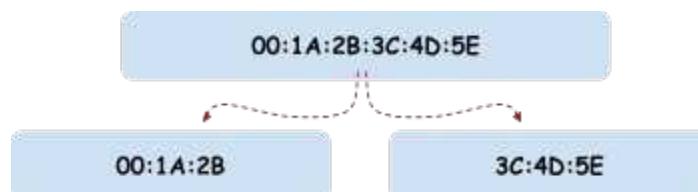
- حجم الـ Interface ID عبارة عن 64 بت وحجم الـ MAC Address عبارة عن 48 بت وبالتالي باقي

16 بت بتمثلهم بـ FF-FE

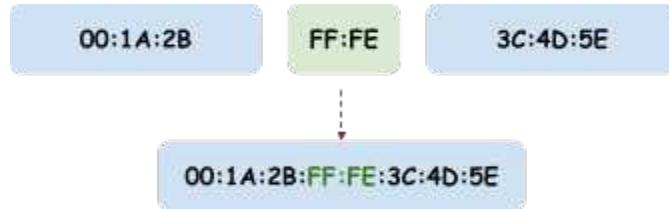
- مثلاً معانا الـ MAC التالي

00:1A:2B:3C:4D:5E

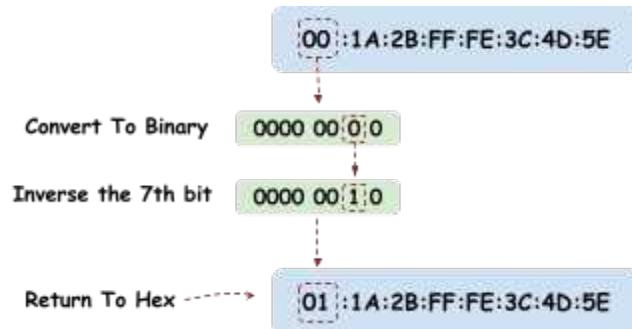
- الخطوه الاولى تقسيم عنوان MAC إلى نصفين



- الخطوة الثانية هي إضافة قيمة ثابتة عبارة عن 16 بت وهي FF:FE بين الجزئين، ليصبح العنوان:



- الخطوة الثالثة: عكس قيمة البت السابع من اليسار



- آخر خطوة هي استخدام الـ Prefix الخاص بالـ Link-Local وهو FE80::

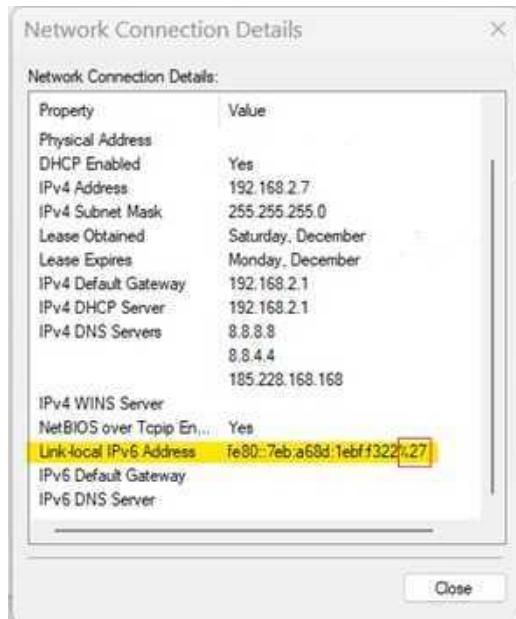


الطريقة دي مستخدمة في Windows XP و Cisco Routers وبعض الانظمة الأخرى .. لكن ابتداءا من Windows 7 تم تغيير الطريقة دي على انظمة Windows وخلت الـ Host ID يبقى Random. اما الـ Network ID زى ما هو FE80::.

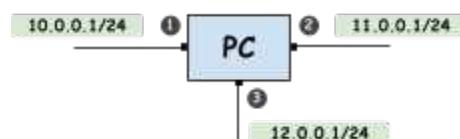
وقبل ما الجهاز يثبت الـ Host ID سواء كان عشوائي او المشتق من الـ MAC، بيعت لكل الاجهزه اللي في الشبكة رسالة Neighbor Solicitation عشان يعرف في جهاز واحد نفس الـ ID ولا لا .. لو مفيش جهاز واحد نفس الـ ID >> بياخد الـ IP وبيعد للاجهزة عشان ياكد عليهم انه اخد الـ ID دا .. والعملية دي اسمها Duplicate Address Detection - DAD.

4.4. IPv6 Zone Identifier

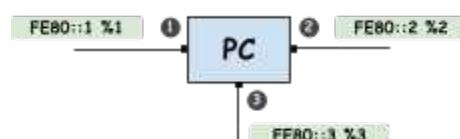
لو فتحت الـ Network Adapter من الـ Control Panel ، هنالكي مكتوب بجانب IPv6 علامة % ورقم .. في IPv6، تستخدم علامة النسبة المئوية % للإشارة إلى Scope ID أو Zone Identifier، وهو مفهوم يستخدمScoped لتحديد الواجهة (Interface) أو النطاق (Scope) عند التعامل مع عناوين أو Link-Local .Multicast Addresses



مثلاً في IPv4 لو عملت Ping على IP: 11.0.0.3 على Packets الـ <> الـ Network Adapter رقم 2 لأنه في نفس الشبكة



اما في IPv6 بنحدد الـ Zone Identifier بناء على الـ Interface



وبالتالي عشان تعمل Ping على Link Local IP معين >> لازم تحدد الوجهة

مثال:

في Linux بنحدد باسم الـ Interface

```
> ping FE80::1%eth0
```

اما في Windows بنحدد بالـ Index ID الخاص بالـ Interface

```
> ping FE80::1%5
```

لعرض كل الـ Interfaces والـ Index ID الخاص بيها، اكتب الامر التالي في الـ CMD

```
> netsh interface ipv6 show interfaces
```

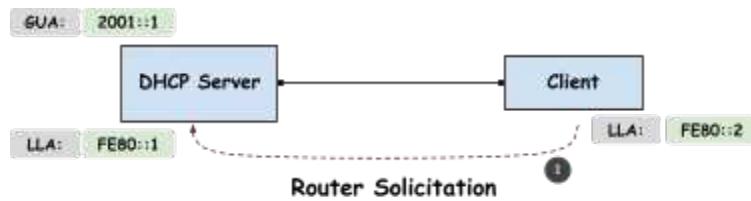
الـ IP local Link شبيه بالـ APIPA في IPv4 •

لو الجهاز لقط IP Dynamic، مش بيفقد الـ IP local •

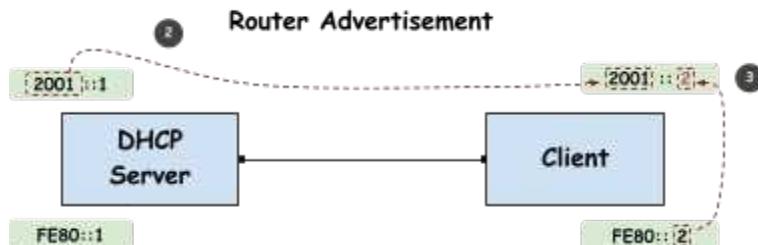
4.5. SLAAC

في IPv4 لازم يكون في DHCP Server شغال ومهياً عشان تعرف تاخد Dynamic IP <> لكن في IPv6 ممكن ال Router يوزع ال Network ID حتى لو مش مشغل عليه خدمة ال DHCP Server .

- اول حاجة ال Client بيكون معاه Unroutable IP وهو ال Link Local Address
- ال Client بيبيعت Router Solicitation (RS) Message او ما يسمى بـ Globally Unique Address - GUA



ال Router Advertise (RA) Message بـ Client فيها ال Prefix ID فيه ريد على ال Router •



في عملية ال SLAAC، ال Router يوزع Gateway IP و Network ID فقط .. بس عشان ال Client يقدر يطلع انترنت لازم يكون معاه DNS وبالتالي ممكن استخدم حلين:

- ممكن اكتب ال DNS بشكل يدوي
- او افعل خدمة ال DHCPv6 Server واللي فيها نوعين:
 - Stateless DHCPv6
 - Random MAC Generate Client يقدر يعمله بنفسه سواء من ال MAC او Interface ID
 - Stateful DHCPv6 .. وهنا ال Router يوزع كل حاجة زي DHCPv4 او

4.6. IPv6 Addressing

Global Unique Address (GUA)

المقابل لـ Public Address في IPv4 •

يتميز بـ 3 بتات من اليسار عبارة عن 001 •

وبما أن كل رقم عبارة عن أربعة بت، فممكن يكون •



كذا لو الـ IP بالـ Hexa يبدأ بـ 2 او 3 يعني •

Link Local Address (LLA)

هو المقابل لـ APIPA في IPv4 ويبدأ بـ FE80::/10

1111 1110 10 00	x → FE 8X
1111 1110 10 01	x → FE 9X
1111 1110 10 10	x → FE AX
1111 1110 10 11	x → FE BX

Site Local

دا كان المقابل لـ Private IP وتم الغاؤه .. والـ Documentation بتاعته مش واضحة .. يبدأ بـ FEC0::/10

Unique Local

المقابل لـ Private .. ويبدأ بـ FC00::/7

Loopback 1/128::

المقابل لـ Loopback في IPv4 (127.0.0.1) .. وهو عبارة عن عنوان واحد فقط ::1/128

Multicast FF00::/8

Unspecified

0.0.0.0 in IPv4 and ::0/128 in IPv6

4.7. Address Grouping

- Unicast
- Multicast FF00::/8

اشتقو من الـ Range دا اتنين IP

- FF02::1 For all nodes or devices in the network
- FF02::2 For all Routers → 224.0.0.2 in IPv4

- Anycast

From one to nearest

لو حطيت الـ IP بطريقة anycast <> ممكن احط نفس الـ IP لاكتر من Router او Server بحيث لو

واحد وقع، الثاني يكون شغال

4.8. IPv6 Header



الـ Flow Label او Flow Control بيستخدم لو فاتح اكتر من Tab وكل Tab بيكون لها رقم محدد .. •

مش كل الاجهزه بتدعمها ومش موجود زيه في IPv4

الـ Total Length هو المقابل للـ Payload Length •

الـ Next Header بيحدد ايه المعلومات اللي هتحط بعد الـ Header •

- الـ TTL في IPv4 هو المقابل للـ Hop Limit في IPv6
- في حجات تانية زي الـ Header Checksum والـ Fragment Offset ودول بيتتحطوا في IPv6 كـ Next Header والـ Extension اللي بيشاور عليهم الـ Header
- حجم الـ Header في IPv6 ثابت ويبلغ 40 بايت على عكس IPv4، حجم الـ Header يتغير بسبب بعض الحقول الاختيارية.

5. Cables

الكابلات دورها توصيل الأجهزة ببعضها، سواء كانت أجهزة كمبيوتر، سويتشات، راوترات، أو حتى Servers اختيار الكابل المناسب بيأثر بشكل مباشر على أداء الشبكة وسرعتها وكفاءتها. وبالتالي لازم تكون فاهم أنواع الكابلات، واستخدام كل نوع.

أنواع الكابلات في الشبكات

5.1. Coaxial Cable

كان منتشر في الماضي لكن نادر الاستخدام حالياً في الشبكات المحلية. وهو عبارة عن سلك نحاس متغطي بمادة بيضاء من الفوم وحوليه Aluminum Shield للتقليل من الـ - Electromagnetic Interference EMI واخر طبقة من البلاستيك.



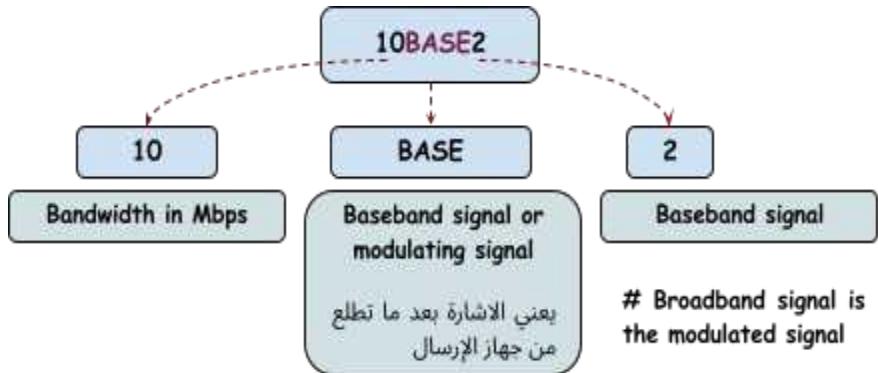
أنواعه:

Srs: كابل سميك يُستخدم في الشبكات القديمة. أقصى مسافة من الـ Thicknet - 10BASE5 •

الـ Dest لا تزيد عن 500 متر

Srs: كابل أنحف وأقل تكلفة، لكنه غير مستخدم حالياً. أقصى مسافة من الـ Thinnet - 10BASE2 •

الـ Dest لا تزيد عن 185 متر



- RG-59 و RG-6: يستخدموا في توصيل الإنترنت والتلفزيون.

ملحوظة:

- لو تعددت اقصى مسافة، بنحط Repeater عشان يقوى الاشارة.
- الـ Bandwidth هو اقصى سرعة لمرور البيانات داخل الشبكة
- الـ Throughput هو السرعة الفعلية لمرور البيانات داخل الشبكة
- لما بنتكلم عن نقل البيانات داخل الشبكة دايما بنتكلم بوحدة bit _bps .. اما في الكمبيوتر بنتكلم بوحدة Bps < Byte

- الـ BNC Connector الخاص بالـ Coaxial Cable اسمه Connector
- بنستخدم في آخر الـ Coaxial Cable حاجة اسمها BNC Terminator عشان تمنع ارتداد البيانات
- لتوصيل اتنين Coaxial Cable بعض بنستخدم BNC-T Connector



5.2. Twisted Pair Cables

عبارة عن أربع أزواج من الأسلاك النحاسية .. كل سلك ملفوف حوليه سلك تاني عشان يعمل مجال مغناطيسي عكسي ويقلل الدا EMF بشكل كبير والخاصية دي اسمها Crosstalk. وفي منه نوعين:

5.2.1. Unshielded Twisted Pair - UTP

هو الأكثر استخداماً في الشبكات المحلية (LAN). وميزة انه رخيص وسهل الاستخدام بس عيبه ان اقصى مسافة يتحملها 100 متر.



هنلاحظ ان كل سلك له لون معين وملفوف حواليه سلك من نفس اللون .. مثلا ازرق وابيض-فارزق

5.2.2. Shielded Twisted Pair - STP

نفس فكرة الدا UTP لكن مع طبقة إضافية من الدا Aluminum Shield على كل زوج من الأسلاك للحماية من الدا EMI وفي بعض الأنواع في طبقة خارجية على جميع الأسلاك.



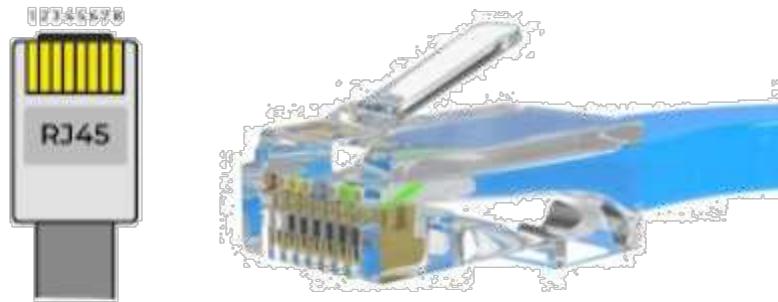
أنواع الـ Twisted-Pair Cables

Category	Maximum Bandwidth	Maximum Data Rate	Maximum Distance Supported	Common Applications
Cat1	0.4 MHz	1 Mbps	—	Telephone and modem lines
Cat2	4 MHz	4 Mbps	—	Telephone
Cat3	16 MHz	10 Mbps	100 meters	10Base-T Ethernet
Cat4	20 MHz	16 Mbps	100 meters	Token ring
Cat5	100 MHz	100 Mbps	100 meters	100Base-T Ethernet
Cat5e	100 MHz	1 Gbps	100 meters	100Base-T Ethernet Home use
Cat6	250 MHz	1 Gbps	100 meters 37 meters for 10 Gb data rates	Gigabit Ethernet Commercial establishments
Cat6a	500 MHz	10 Gbps	100 meters	Gigabit Ethernet Enterprise data centers Commercial establishments
Cat7	600 MHz	10 Gbps	100 meters	10 Gbps core infrastructure
Cat7a	1,000 MHz (1 GHz)	10 Gbps	100 meters 50 meters for 40 Gb data rates	10 Gbps core infrastructure
Cat8	200 MHz (2 GHz)	Cat8.1: 25 Gbps Cat8.2: 40 Gbps	30 meters	25/40 Gbps core infrastructure

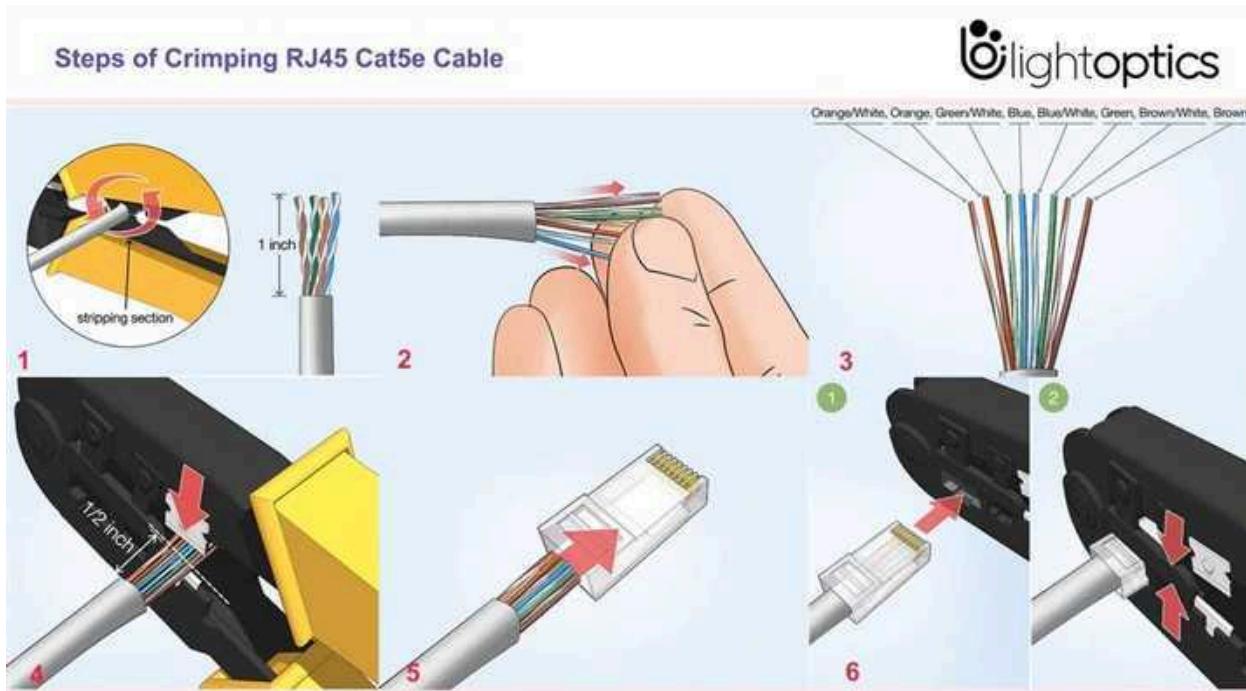
- اختصار لـ Category .. حرف الـ e يعني اصدار محسن .. حرف الـ a يعني اصدار "Enhanced" .. حرف الـ e اختصار لـ "augmented" يعني اصدار معزز
- Cat1 و Cat2 اصدارات قديمة جداً ومش مستخدمة دلوقتي .. Cat4 كان بيستخدم مع شبكات الـ Token Ring القديمة والتي مش مستخدمة دلوقتي برضو.
- حرف الـ T في 100Base-T مثل معناه انه Twisted .. وفي اصدارات تانية بتكون TX ودا معناه انه يدعم Full-duplex و Half-duplex
- لو المسافة تعدد الحد الاقصى، بنضيف Switch
- تفاصيل اكتر عن كل نوع هنا:

<https://telecom.samm.com/history-of-ethernet-lan-cables-categories>

الـ RJ-45 Connector اسمه Twisted Pair Cables بتستخدم



وعشان نوصل الـ RJ45 Connector بالكابل، بنسخدم Cable Stripper عشان نشيل الطبقة الخارجية من على الأسلاك وبعدين نستخدمو الـ Wire Crimper لتوصيل الأسلاك بالـ Connector. بس العملية دي بتحتاج ترتيب معين للأسلاك عشان نوصلها بطريقة صحيحة بين الأجهزة المختلفة.



5.2.3. Clipping Types

هي طريقة توصيل الـ Ethernet Ports الخاصة بالأجهزة مع بعض باستخدام الـ Twisted Pair Cables ويتختلف على حسب نوع الجهاز .. وفي عدنا نوعين

- أجهزة من نوع MDI وهي اختصار لـ Media Dependant Interface

غالباً يتبع في الـ Routers زي الـ PCs, laptops, servers, IP phones وكمان الـ End Device والـ Network Printers.

- أجهزة من نوع MDI-X وهي اختصار لـ Media Dependant Interface Crossover زي الـ Switch والـ Hub.

الفرق بينهم هو ان كل نوع يحدد Pins مختلفة لارسال البيانات واستقبالها .. يعني الأجهزة من نوع MDI تتبع على Pin 1,2 Pin 3,6 وتستقبل على Pin 1,2 .. اما أجهزة MDI-X تتبع على Pin 3,6 Pin 1,2 وتستقبل على

في 3 طرق للتوصيل

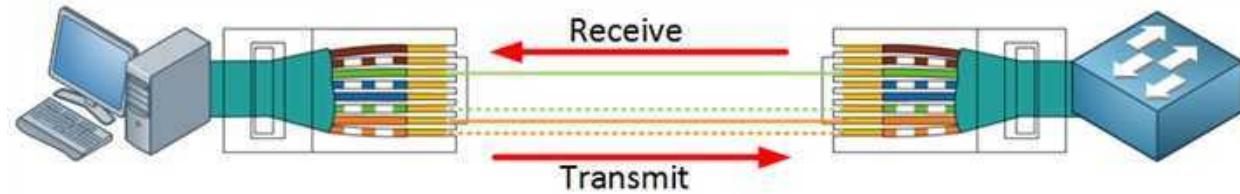
- (MDI to MDI-X) Straight-through للتوصيل بين جهازين من نوع مختلف

- (MDI-X - MDI-X) Crossover للتوصيل بين جهازين من نفس النوع (MDI - MDI)

- Rollover

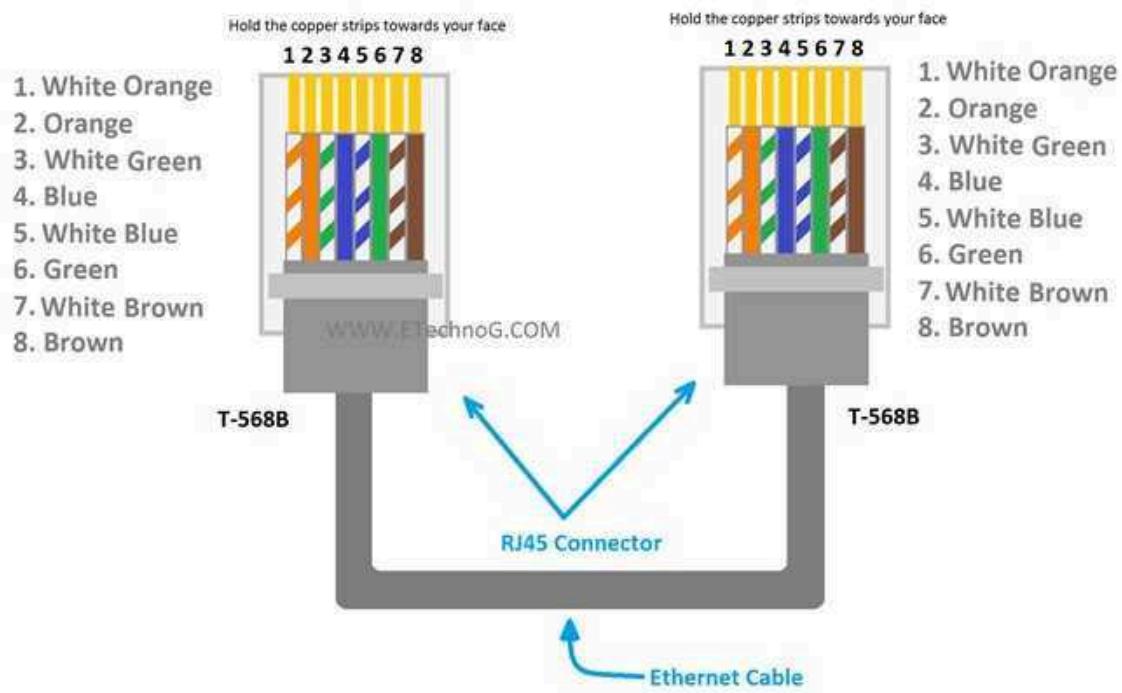
كرات الشبكة سواء T - 10BASE-T او Fastethernet او Gigethernet تستخدم أربعة Pins للارسال والاستقبال على حسب نوع الجهاز زي ما قلنا وبقي الـ Pins او الأسلاك مش مستخدمين الا لو الأجهزة بدعم الـ PoE وفي الحالة دي يتم استخدامهم لتزويد الجهاز بالطاقة بدلاً من الاعتماد على Adapter لكل جهاز .. اما الـ 8 أسلاك ويقدر يرسل او يستقبل على اي سلك.

Straight Through Cable

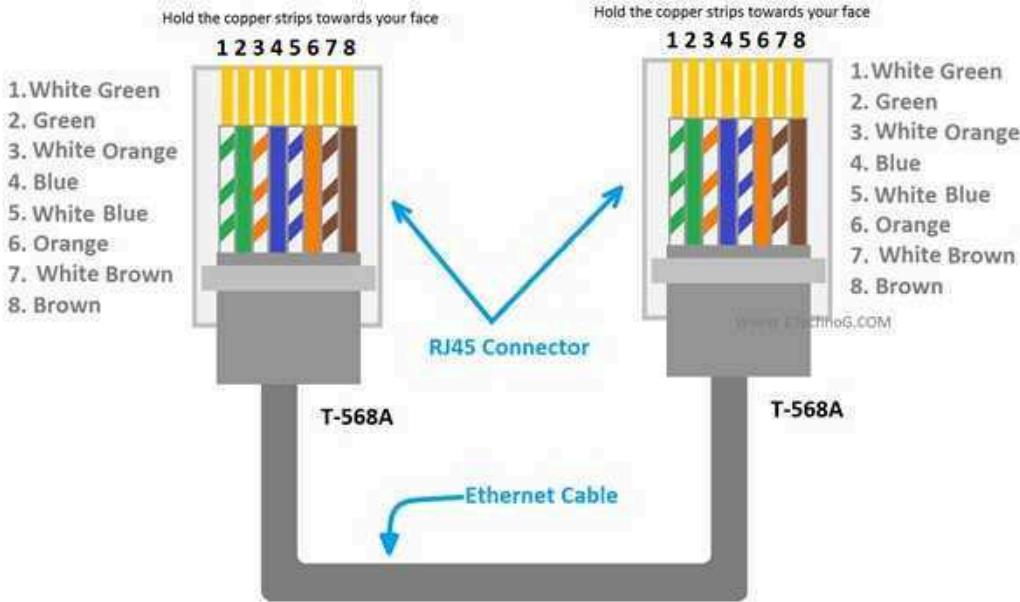


- الألوان الموضحة في الصورة وترتيبها هي الـ Standard والتي حددتها منظمة TIA/EIA، والافضل تمشي على الـ Standard عشان غالبا اللي بيوصل الطرفين مش نفس الشخص.
- ولكن لو مش هتمشي على الـ Standard اهم حاجة نوصل Pins 1,2,3,6 بنفس الـ Pins عند الطرف الثاني وترتيب الألوان مثلًا لو Pin1 لونها بني بقى Pin2 لونها بني-أبيض وهكذا والطرف الثاني لازم يكون نفس ترتيب الألوان .. لأن كل لون بيبقى بنفس الترتيب في الطرفين في الـ Straight Through
- الـ Standard دا من ترتيب الألوان اسمه T-568B

صورة أوضح لشكل الكابل



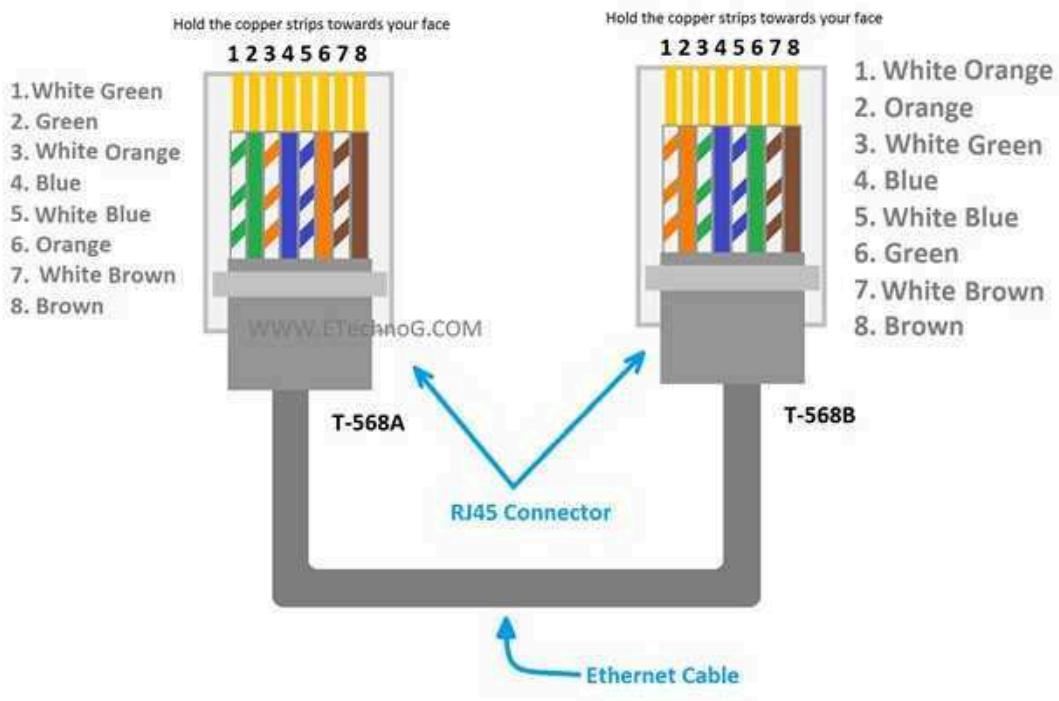
صورة اخرى بتوضح Standard T-568A



Crossover



- برضو الترتيب الموضح في الصورة هو الـ Standard .. ولكن بما ان الاجهزه من نفس النوع .. يعني بيعutto يستقبلو على نفس الـ Pins فـ هنعكس ارقام الـ Pins لو استخدمنا نفس ترتيب الالوان باقي الـ Pins بنوصلهم Straight عشان لو هنسخدمهم في نقل الطاقة PoE



الأجهزة الحديثة دلوقتي بتدعم Auto-Mdi X يعني لو وصلتها باي طريقة من الاتنين هو هيعمل Detect نفسه ويحدد الارسال والاستقبال هيكون على اي Pins

Rollover (Console)

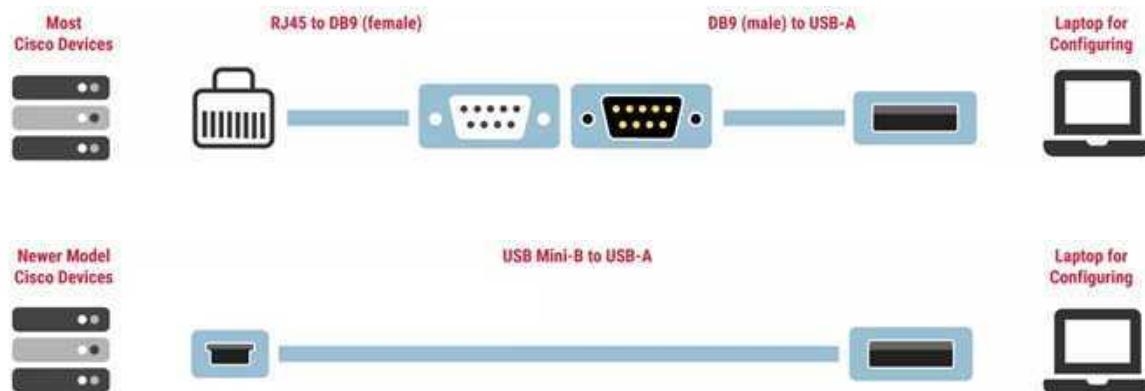
كابل الـ *Cable* هو الـ *Cable* اللي بيستخدم لتوصيل الـ *Computer* بالـ *Network Device* عشان تعملو *Console Port* .. وله اكتر من شكل بس الأشهر بيكون عبارة عن طرف *RJ45* بيتوصل في الـ *Configuration* على الـ *Network Device* والطرف الثاني او اسمه *Serial* ممكن توصلو مباشرة بالـ *Computer*, غالباً بنوصلو بـ *USB Adapter* للتحويل من *Serial* الى *USB*

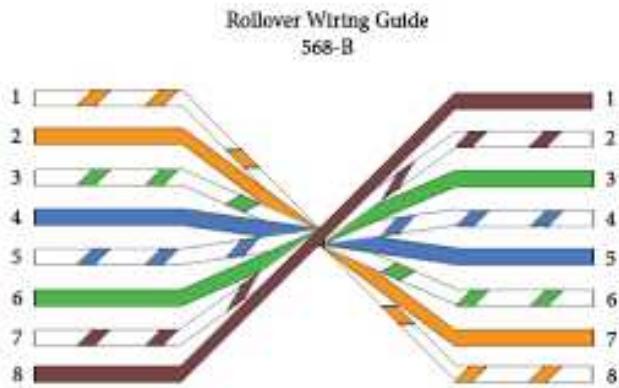
شكل الـ *RJ45 to DB9 - Console Cable*



وفي انواع تانية

- *USB A to Mini-B*
- *USB to RJ45*





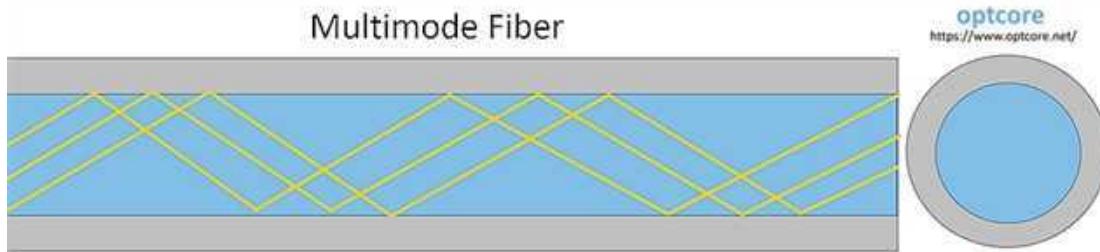
5.3. Fiber Optic Cables

كابلات الألياف الضوئية (Fiber Optic Cables)

- عبارة عن سلك رفيع من الزجاج او من الـ Plastic في بعض الأنواع الرديئة، ومغطى بطبقة من الـ Clading للحفاظ على انعكاس الضوء داخل الـ Cable ثم طبقة اخرى من البلاستيك.
- أسرع وأكفاء من الكابلات النحاسية، ويتستخدم الضوء بدلاً من الكهرباء لنقل البيانات. عن طريق إرسال نبضات او Pulses من الضوء بدلاً من الـ Zeros والـ Ones.
- مناسبة للمسافات الطويلة والشبكات عالية السرعة زي شبكات الـ WAN والـ Datacenters.
- في منه نوعين:
 - سميكة الزجاج اكبر ويسمح بمرور ضوء اكتر ولكن يدعم مسافات صغيرة
 - سميكة الزجاج صغير وبالتالي يسمح بمرور ضوء اقل ولكن يدعم مسافات اكبر

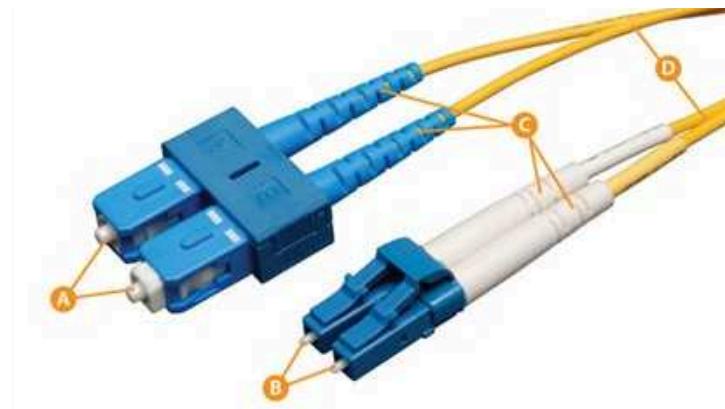


شكل انكسار الضوء داخل الـ Fiber Optic Cable



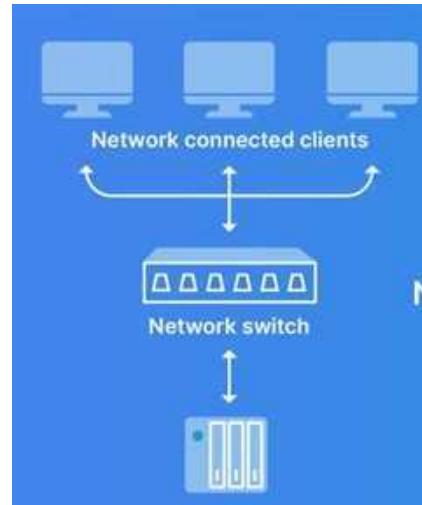
الـ Multimode يدعم مرور اكتر من شعاع من الضوء

شكل الـ Fiber Optic Connectors



في انواع بتدعم الـ Duplex Mode، وبالتالي الكابل يبقى في اثنين Connectors .. واحد للإرسال وواحد للاستقبال .. وفي انواع حديثة بتدعم الإرسال والاستقبال على Cable واحد اسمها **Fiber**

الـ Fiber Optic Cables بتستخدم كمان في الـ Storage Area Network بسبب سرعتها العالية جدا، وال SAN عبارة عن Array او Rack يركب فيها مجموعة من الـ Hards .. ويتوصل بـ Switch والـ Server متصل بالـ Servers بتاعت الشركة .. التوصيلات بين السويتش والـ SAN وبين الـ SW وبقى الـ Fiberoptic تكون من الـ



6. Configuration Modes

عند التعامل مع أجهزة Cisco، في أكثر من طريقة للوصول للجهاز (سواء كان راوتر أو سويفت أو ... Firewall). كل طريقة لها غرض معين ومستوى أمان مختلف. الطرق دي بتسمى **Access Types** وبتشمل الآتي:

6.1. Configuration Types

- **CLI**

Using Command Line, More Suitable and More Secure

- **CLI (Console Access)**

دي الطريقة الأساسية للوصول للجهاز مباشرة من خلال كابل الـ Console اللي اتكلمنا عنه واللي بيكون RJ45-Serial عادة .. وبنستخدم الطريقة دي للوصول المحلي للجهاز لما يكون لسه جديد أو مش متصل بالشبكة.

وعشان نستخدم الطريقة هنحتاج كابل Console، وبرنامج Terminal Emulator (زي Hyper Terminal و SecureCRT أو PuTTY و Tera Term). وهي تطبيقات مفتوحة المصدر أو ودي تطبيقات مدفوعة واخيرا Mobaxterm ودا الأفضل لنظام لينكس).

باقي الطرق بتحتاج نعمل Configuration الاول عشان نهیں الراوتر لاستخدامها.

- **Telnet Access**

بروتوكول للوصول للجهاز عن بعد باستخدام CLI عبر الشبكة. والطريقة دي مش بتشفير التрафيك، وبالتالي مش آمنة للاستخدام في الشبكات الحساسة.

- **SSH Access (Secure Shell)**

بروتوكول مشفر وآمن للوصول عن بعد للجهاز. يُفضل استخدامه بدل Telnet.

- **GUI**

More easier and faster, but more vulnerable and not stable as CLI.

- **Web Access (HTTP/HTTPS)**

الوصول لإعدادات الجهاز من خلال واجهة رسومية باستخدام المتصفح.

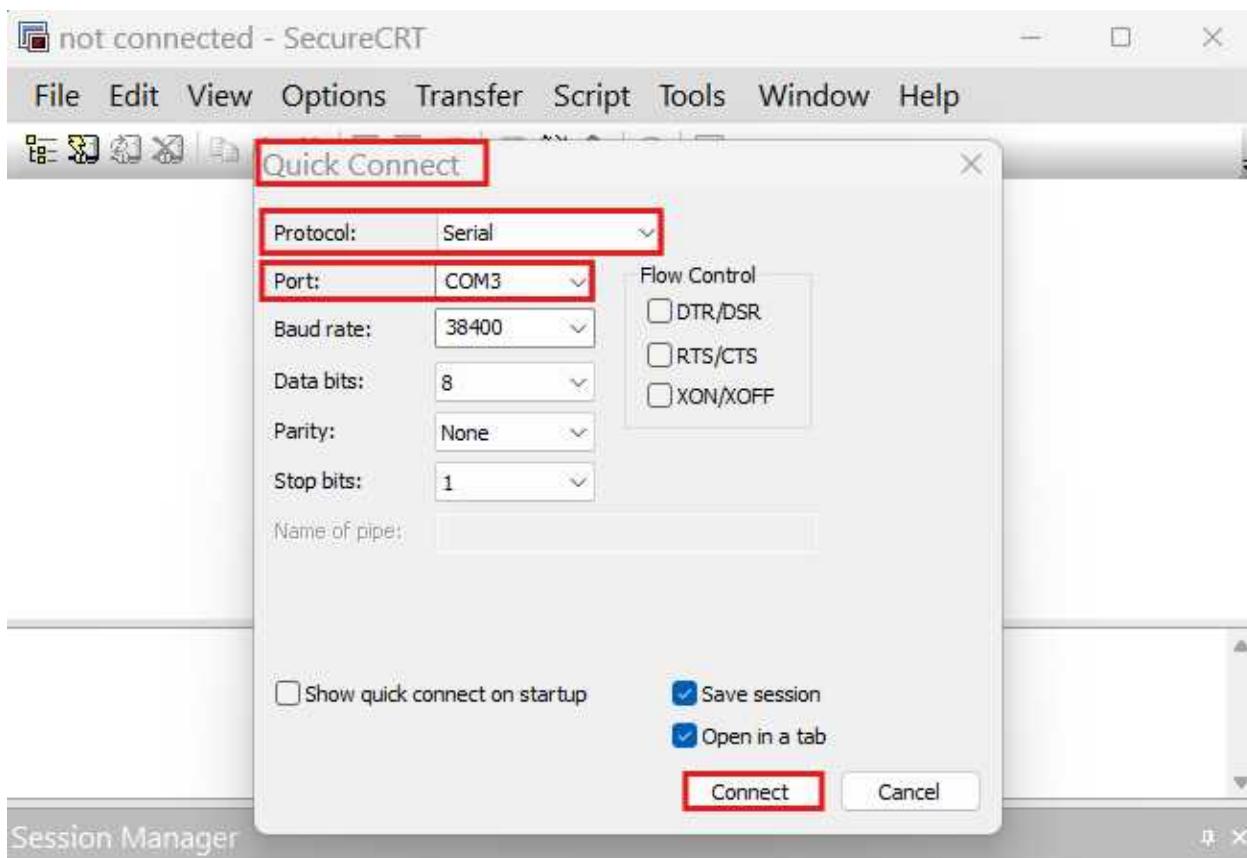
بعد توصيل الlaptop مثلًا بال Router عن طريق كابل Console واستخدام اي Terminal Emulator عشان

نقدر نعمل Configuration للراوتر:

افتح الـ Terminal Emulator مثلًا SecureCRT •

من File اضغط على Quick Connect •

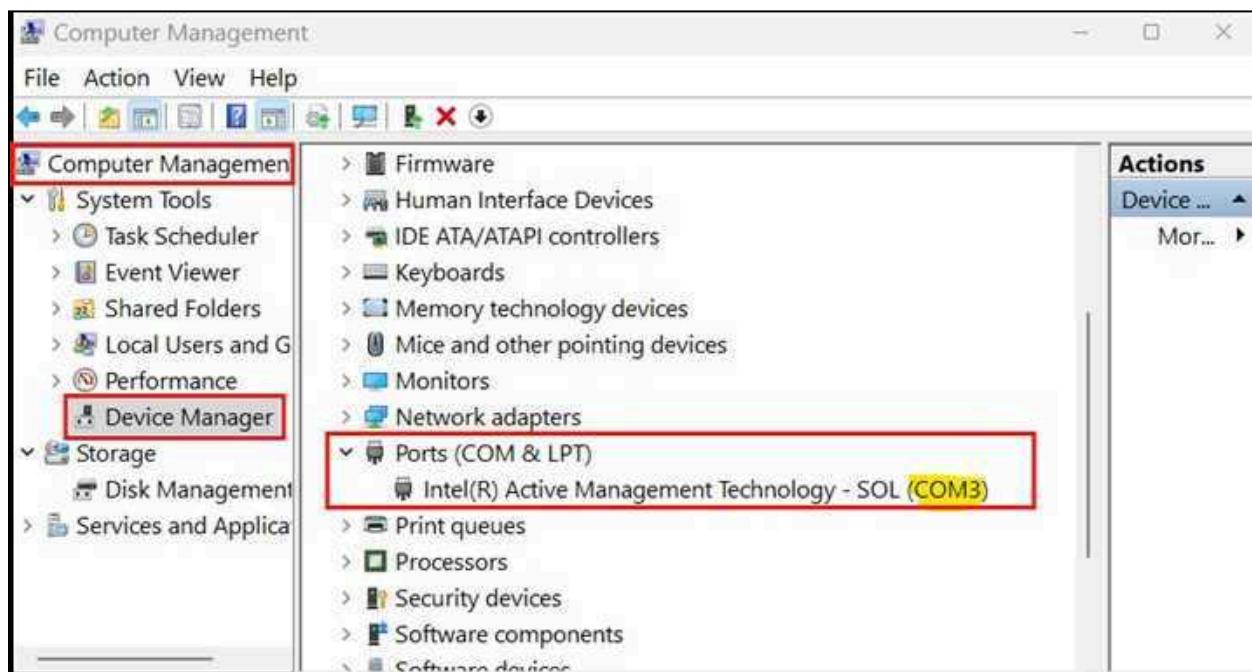
وبعددين اختيار Serial من نوع الـ Protocol واختار اسم الـ Serial Port الموجود على جهازك •



لمعرفة اسم الـ Serial Port

• اضغط على Right Click ثم اختار This Computer

• من Device Manager ثم Ports الـ Ports اللي عندك هتلاقى أسماء الـ Ports



- بعد ما تعمل Connect على الراوتر هيقابلنا اتنين Modes
- في الـ Setup Mode وهو عبارة عن مجموعة من الاسالة لاعداد الـ Configuration الاولية للجهاز
 - وفي الـ Command Mode ودا الـ Mode اللي بنستخدم فيه الـ Commands عشان نعمل الاعدادات المختلفة والمتقدمة

The screenshot shows a Windows application window titled "router". The menu bar includes File, Edit, View, Options, Transfer, Script, Tools, Window, and Help. The toolbar contains icons for file operations like Open, Save, Print, and Copy/Paste. A tab labeled "router" is selected. The main window displays system information and a configuration dialog.

```

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/ww1/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

Cisco 3660 (R527x) processor (revision 1.0) with 187392K/9216K bytes of memory
Processor board ID FTX0945W0MY
R527x CPU at 250MHz, Implementation 40, Rev 1.2, 512KB L2 Cache

3660 Chassis type: ENTERPRISE
2 FastEthernet interfaces
DRAM configuration is 64 bits wide with parity enabled.
253K bytes of NVRAM.
8192K bytes of processor board system flash (Read/Write)

--- System Configuration Dialog ---

would you like to enter the initial configuration dialog? [yes/no]:
% Please answer 'yes' or 'no'.
would you like to enter the initial configuration dialog? [yes/no]:

```

- اول مرة تعمل Access للجهاز هيسالك عايز تعمل الـ Initial Configuration ولا لا
- لو كتبت Yes هيدخلك على الـ Setup Mode
 - لو فتحت الـ Command Mode وعايز تطلع للـ Setup Mode اضغط Ctrl+C
 - لو كتبت No هيدخلك على الـ Command Mode

6.2. Configuration Modes

في نظام التشغيل الخاص بأجهزة Cisco (زي الـ IOS)، الـ **Command Line Mode** (زي الـ IOS) يتكون من أكثر من أوضاع مختلفة لإجراء تغييرات على إعدادات الجهاز. كل وضع يختلف في مستوى الصلاحيات ونوع الأوامر التي يمكن استخدامها.

- **User Exec Mode** هو الـ **Mode** الأول

- يتيح تنفيذ أوامر بسيطة واستطلاعية فقط. زي Ping و Show.

- شكل الـ **Prompt** يكون اسم الجهاز متبوع بعلامة ">" ..

Router>

- وممكن تكتب علامة استفهام عشان تعرض جميع الأوامر الموجودة في الـ **Mode** دا

```
Router> Router>?
Exec commands:
access-enable      Create a temporary Access-List entry
access-profile     Apply user-profile to interface
call               Voice call
clear              Reset functions
connect            Open a terminal connection
crypto             Encryption related commands.
disable            Turn off privileged commands
disconnect         Disconnect an existing network connection
enable             Turn on privileged commands
exit               Exit from the EXEC
help               Description of the interactive help system
lat                Open a lat connection
lock               Lock the terminal
login              Log in as a particular user
logout             Exit from the EXEC
modemui           Start a modem-like user interface
mrinfo            Request neighbor and version information from a multicast
router
mstat             Show statistics after multiple multicast traceroutes
mtrace            Trace reverse multicast path from destination to source
name-connection   Name an existing network connection
--More--
```

• تاني Mode هو الـ **Enable Mode**

ده الوضع اللي بيتيح الوصول للأوامر المتقدمة واستعراض إعدادات الجهاز.

- للدخول الى الـ Enable Mode اكتب الامر `enable` او اختصارا `en`
- شكل الـ Prompt بيكون اسم الجهاز متبوع بعلامة `#`

```
Router> enable  
Router#
```

- الوضع دا مهم لأن فيه امر `Reload` اللي بيعمل اعدة تشغيل للجهاز .. والامر دا خطير في بيته العمل لانه ممكن ياثر على الشبكة كلها على حسب مكان الرووتر .. عشان كدا غالبا بيكون في دا للوصول للوضع `Password`

• الـ **Global Configuration Mode**

الوضع اللي بيتمكنك من تعديل الإعدادات العامة للجهاز أو الدخول إلى الـ **Subconfiguration Modes**.

- للدخول الى الـ Configuration Mode اكتب الامر `configure terminal` او اختصارا `conf t`
- شكل الـ Prompt بيكون اسم الجهاز متبوع بـ `(config)`

```
Router# configure terminal  
Router(config)#
```

- الـ **Sub Configuration Mode** بيختلف على حسب الحاجة اللي عايز تعملها .. مثلا في وضع خاص بإعدادات الـ `Interface` ووضع خاص بإعدادات الـ `Routing Protocol` معين، وهكذا.

ملحوظة:

- تقدر تكتب اي حرف من الا Command وبعدين تضغط على Tab للاكمال
- ممكن تكتب اي حرف من أي أمر وبعدين تكتب علامة استفهام لعرض التكملة او لعرض باقي الأوامر اللي بتبدأ بنفس الحروف او باقي الا Options لنفس الأمر

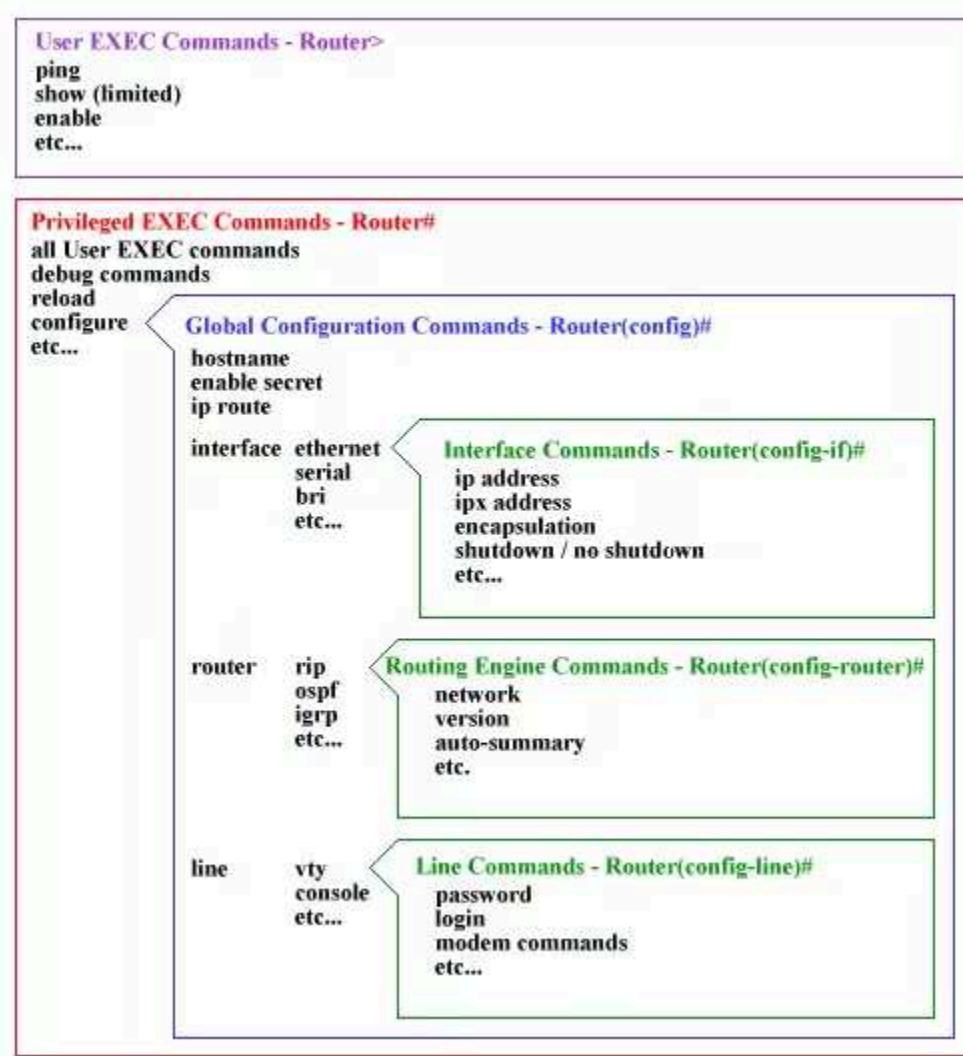
هنا مثلا مكتوب فالآخر -- يعني في تكملة للأوامر المعروضة ، وممكن تضغط Enter لعرض الأمر السطر التالي او Space لعرض صفحة الأوامر التالية او q للخروج وعرض الا Prompt

```
Router> Router>?
Exec commands:
access-enable      Create a temporary Access-List entry
access-profile     Apply user-profile to interface
call               Voice call
clear              Reset functions
connect            Open a terminal connection
crypto             Encryption related commands.
disable            Turn off privileged commands
disconnect         Disconnect an existing network connection
enable             Turn on privileged commands
exit               Exit from the EXEC
help               Description of the interactive help system
lat                Open a lat connection
lock               Lock the terminal
login              Log in as a particular user
logout             Exit from the EXEC
modemui            Start a modem-like user interface
mrinfo             Request neighbor and version information from a multicast
router             Router
mstat              Show statistics after multiple multicast traceroutes
mtrace             Trace reverse multicast path from destination to source
name-connection    Name an existing network connection
--More--
```

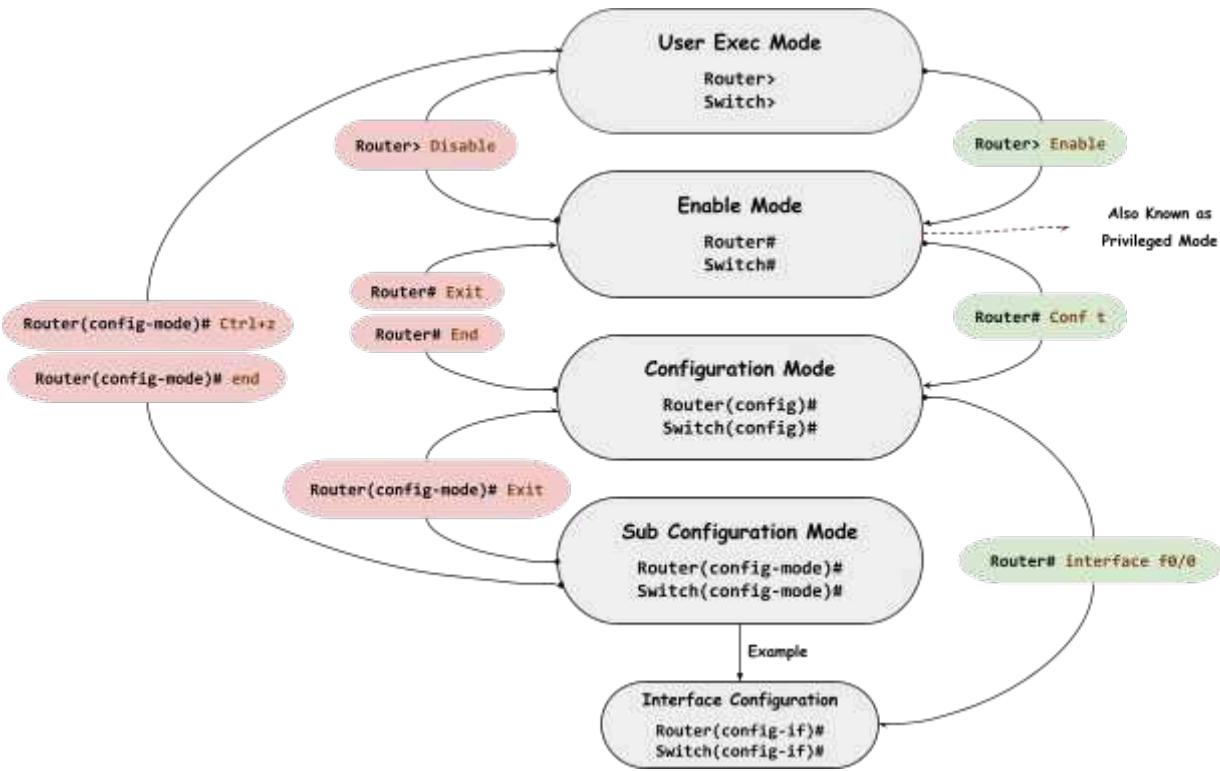
- لكتابة .. اكتب علامة تعجب ملحوظة بالتعليق اللي انتا عايزو

```
Router# ! this is a comment
```

الصورة التالية بتوضح كل Mode واسهر الاوامر الموجودة فيه



الصورة التالية بتوضح طريقة الانتقال بين الـ Modes المختلفة



عند بدء إعداد أي جهاز راوتر أو سويفت من Cisco لأول مرة، هناك مجموعة من الإعدادات الأساسية التي يجب تطبيقها لضمان تشغيل الجهاز بكفاءة وأمان. الإعدادات دي بتتضمن تكوينات للـ **Hostname**، وتأمين الوصول، وتمكين البروتوكولات الأساسية.

- تعيين اسم للجهاز لتمييزه داخل الشبكة:

```
Router> enable
Router# configure terminal
Router(config)# hostname R1
```

- لضبط وقت الـ **timeout** عشان تحكم في المدة اللي يفضل فيها الرووتر أو السويفتش منظر قبل ما يقفل الجلسة في حالة عدم النشاط (Idle Session)، الإعدادات دي ممكن تحكم فيها حسب طريقة اتصالك بالرووتر (**Console** أو **VTY**).

لتغيير المدة بالنسبة لـ **Console Port** من خلال الـ **exec-timeout**

```
Router(config)# line console 0
R1(config-line)# exec-timeout <minutes> <seconds>
R1(config-line)# exit
```

رقم 0 في أمر **line console 0** بيمثل رقم الـ **Session** وكدا كدا مفيش غير منفذ وحيد للـ **Console** فبنكتب 0

ممكن تكتب في الدقائق والثواني 0 عشان الجلسة تفضل مفتوحة، بس لا ينصح بالغاء الـ **Timeout** في بيئة العمل.

لتغيير المدة بالنسبة لـ **VTY** من خلال الـ **exec-timeout**

```
Router(config)# line VTY 0
R1(config-line)# exec-timeout <minutes> <seconds>
R1(config-line)# exit
```

- معظم الاوامر ممكن تلغيها عن طريق كتابة **no** قبل الأمر.
- في **Action System Logs** بتظهر لو حصل معين .. وممكن تتدخل مع الامر اللي بتكتبو ومبيقاش واضح. ممكن تستخدم الامر التالي لمنع تقاطع رسائل النظام مع الأوامر أثناء الكتابة.

```
Router(config)# line console 0
R1(config-line)# logging synchronous
R1(config-line)# exit
```

- لو كتبت اي امر بطريقة خاطئة فالـ `Enable Mode` او الـ `User Mode` الراوتر يفترض انك بتكتب اسم جهاز معين عشان توصله من خلال الـ `Telnet` وبيفضل مهنج شوية لو اسم الجهاز مش موجود في الشبكة، لانه بيدور عليه عن طريق الـ `DNS Lookup`.

حل المشكلة دي:

- ممكن تلغي الـ `DNS Lookup` عن طريق الامر التالي

```
R1(config)# no ip domain-lookup
```

- او اضغط على `6 + Ctrl + Shift` .. اتأكد انك ضغط على 6 اللي فوق الـ `Keyboard` مش اللي على اليمين
- حفظ الإعدادات

```
R1# write memory
```

او

```
R1# copy running-config startup-config
```

Copy r s للختصار اكتب

- عرض كل الـ `Configuration` الموجودة

```
R1# show running-config
```

- لكتابة اي امر خاص بالـ `Configuration Mode` داخل الـ `Enable Mode` اكتب `do` قبل الامر

7. Router Passwords

عند إعداد الراوتر لأول مرة، من المهم جدًا ضبط كلمات سر لحمايته من الوصول غير المصرح به. في Cisco او الـ Switches Routers، ممكن نعمل إعداد لكلمات السر على اكتر من حاجة، وحماية جزء معين من الراوتر. هنشرح الأنواع الأساسية مع الأوامر الخاصة بكل واحدة.

أنواع الـ Configuration Passwords اللي ممكن نعملها

- كلمة سر للدخول على الراوتر من خلال الـ Console Cable <_ Console Password <_
- كلمة سر للدخول الى الـ Enable Mode <_ Enable Password <_
- كلمة سر للدخول على الراوتر من خلال SSH او Telnet <_ VTY Password <_
- كلمة سر للدخول على الراوتر من خلال Aux <_ Aux Password <_

لإعداد Console Password

```
Router(config)# line console 0
R1(config-line)# password <your_password>
R1(config-line)# login
R1(config-line)# exit
```

لإعداد Enable Password

```
Router(config)# enable password <your_password>
```

لإعداد Password لتأمين الوصول البعيد للراوتر عن طريق SSH أو Telnet

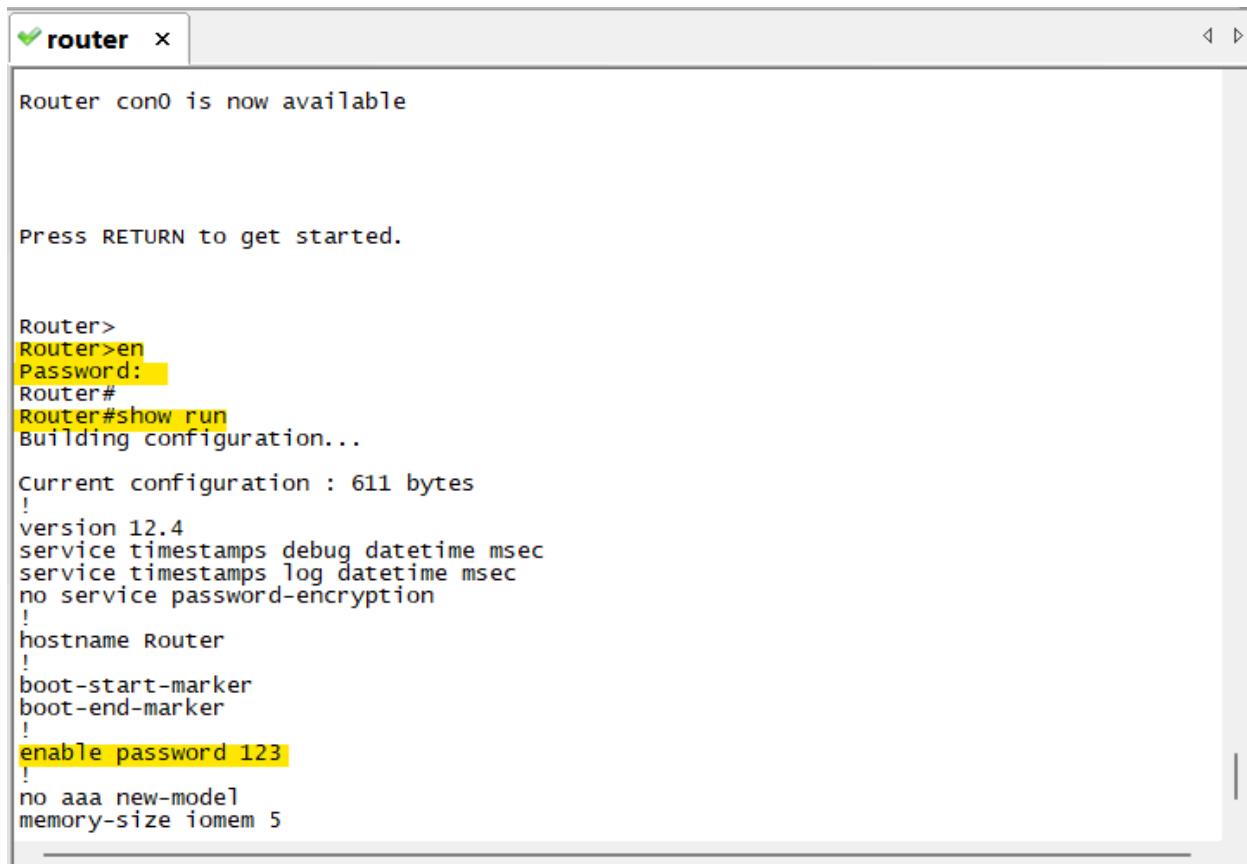
```
Router(config)# line vty 0 4
R1(config-line)# password <your_password>
R1(config-line)# login
R1(config-line)# exit
```

هنا بنكتب من 0 الى 4 يعني بنحدد اربعة Sessions للوصول للراوتر من خلال ssh او telnet

7.1. Password Encryption

الطرق اللي طبقناها لإعداد الـ Password بتخلية يتخزن على الراوتر بشكل Clear-text، وبالتالي هيظهر في الـ Show-Running.

مثلا لو عملت اعداد لكلمة سر 123 على الـ Enable Mode .. وبعدين كتبت Show run لعرض إعدادات الجهاز



A terminal window titled "router" showing the configuration of a Cisco router. The window displays the following text:

```
Router con0 is now available

Press RETURN to get started.

Router>
Router>en
Password:
Router#
Router#show run
Building configuration...

Current configuration : 611 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
enable password 123
!
no aaa new-model
memory-size iomem 5
```

هناقي الامر اللي كتبتو لإعداد Password وإلـ Enable Password واضح

عشان كدا في اكتر من طريقة لتشفيـر الرقم السري

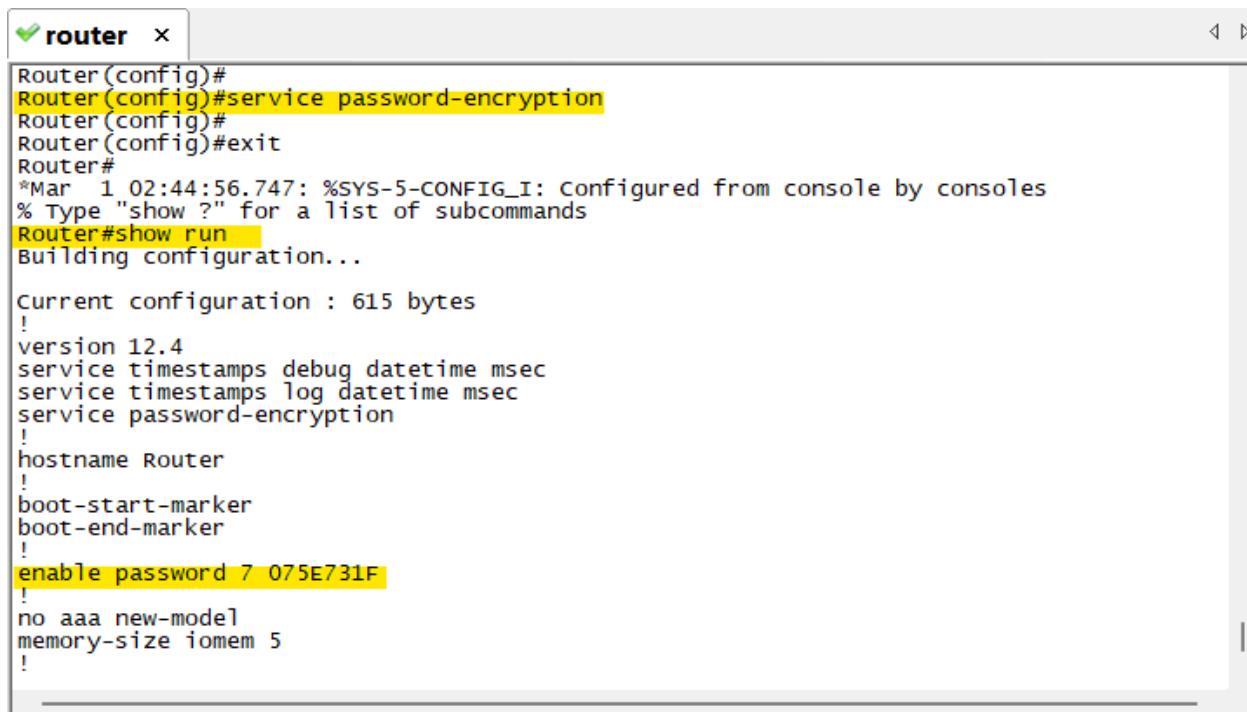
- 7
- Md5 (Message Digest 5)
- SHA (Secure Hashing Algorithm)

لكسر الـ Password المشفر باستخدام md5 او SHA بنستخدم الـ Rainbow Brute Force Attack او Table

لأنه افتراضياً، يتم تخزين كلمات السر في ملف الـ **Running Config** بنص واضح، ممكناً نشفّر كل كلمات السر المخزنة باستخدام **Algorethem 7** عن طريق الامر التالي:

```
Router(config)# service password-encryption
```

هنلاحظ انه تم تشفير الرقم



```
router# Router(config)# service password-encryption
Router(config)#
Router(config)#exit
Router#
*Mar 1 02:44:56.747: %SYS-5-CONFIG_I: Configured from console by consoles
% Type "show ?" for a list of subcommands
Router#show run
Building configuration...
Current configuration : 615 bytes
!
version 12.4
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname Router
!
boot-start-marker
boot-end-marker
!
enable password 7 075E731F
!
no aaa new-model
memory-size iomem 5
!
```

إلغاء الأمر

```
Router(config)# service password-encryption
```

لإعداد Password مشفر بال MD5 أو Line Console او Enable Mode او VTY .. ممكن نستخدم نفس الامر مع استبدال كلمة Password بـ Secret.

- مثال لإعداد Password

```
Router(config)# enable secret <your_password>
```

لو عملت Password عادي وبعدين عملت Secret Password .. الرووتر يحفظ بالاتنين، عشان لو نقلت الـ Configuration لرووتر تاني مش بيدعم الـ Secret Password يقدر يحفظ بالـ Password العادي.

7.2. Privileges Levels

أجهزة Cisco بتستخدم **Privilege Levels** لتحديد مستويات الوصول المختلفة للمستخدمين. وفي 16 مستوى (من 0 إلى 15) يمكن تخصيصهم للتحكم في الصلاحيات، مما يسمح بتأمين الرووتر أو السويفت بناءً على احتياجات الأمان.

في مستويين رئيسيين وهما المستوى الأول User Exec Mode مقصود بيـه الـ Privilege Level 1 ومستوى 15 مقصود بيـه الـ Enable Mode والصلاحيات الكاملة على جميع الأوامر. أما المستويات اللي بين 1 إلى 15 يمكن تخصيصهم.

- Layer Of Security
- تحديد صلاحيات لكل مستخدم (Privileges)
- لمراقبة المستخدمين ومعرفة الأوامر اللي استخدموها (Accounting)

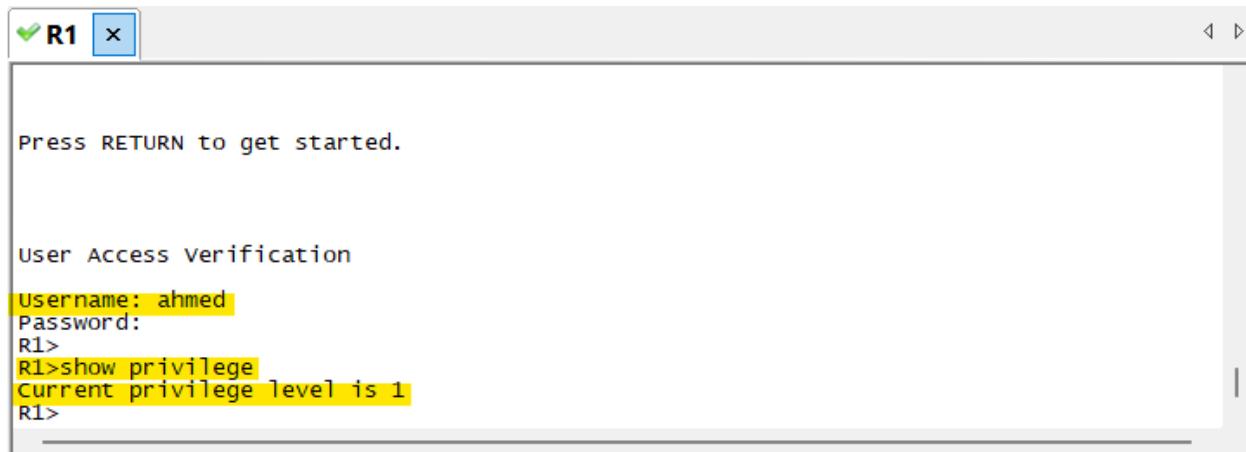
لإعداد Username و Password للدخول على الـ Line Console

```
R1(config)# username ahmed secret 123
```

وبعدين لازم نكتب الامر Local Database في الـ VTY او Line Console عشان يستخدم الـ Login Local .
ممكن نستخدم خارجي زي الـ ACS او ISE Server لعمل Authentication للمستخدمين.

```
R1(config)# username ahmed secret 123
R1(config)# line console 0
R1(config-line)# login local
R1(config-line)# exit
```

الامر دا هيعمل إعداد لمستخدم اسمه Ahmed ورقم سري 123 مشفر بـ Md5 وهيسخدم الـ Local Database By Default عشان يتأكد من المستخدمين اللي داخلين من خلال الـ Line Console .. و المستخدم دا هياخد 1 Privilege يعني اول ما يعمل Login هيدخل على الـ Exec Mode ، وبالتالي ممكن اعمل Mode مختلف على الـ Enable Password عشان المستخدم دا يفضل محدود بالأوامر الموجودة فال Exec Mode



ايه الفرق بين الاتنين:

```
R1(config)# username ahmed secret 123  
R1(config)# line console 0  
R1(config-line)# login local  
R1(config-line)# exit
```

```
R1(config)# line console 0  
R1(config-line)# password cisco123  
R1(config-line)# login
```

- أمر Login بيستخدم Password واحدة عامة (Shared Password) لكل المستخدمين اللي بيعاولوا يدخلوا.

- الباسورد دا بتتحدد باستخدام أمر password على الـ VTY Lines أو Console Line
- مفيش نظام مستخدمين متعدد، يعني أي حد يدخل الباسورد هيخش مباشرة.

لتحديد Privilege Level للمستخدم اللي بيدخل عن طريق Console أو VTY :

```
R1(config)# line console 0  
R1(config-line)# privilege level 15
```

بعد تطبيق الامر دا <> لو حد عمل Login على الراوتر عن طريق الـ Console هيدخل مباشرة على الـ Mode

تخصيص Local Authentication لمستخدم معين عن طريق الـ Privilege Level

```
R1(config)# username admin privilege 15 secret 123  
R1(config)# line console 0  
R1(config-line)# login local  
R1(config-line)# exit
```

لو المستخدم admin عمل Login من خلال الـ Console هيدخل مباشرة على الـ Enable Mode

لتخفيض Privilege Level معين لأوامر معينة:

```
R1(config)# privilege exec level 5 show ip route
```

ال الطبيعي ان اي مستخدم في Level 1 يقدر يستخدم اوامر عرض الإعدادات "show" ولكن بعد تطبيق الامر達 .. هيتم سحب الأمر من Level 1 وتخفيضه لـ Level 5 او اعلى

لتخفيض Privilege Level لـ Password معين

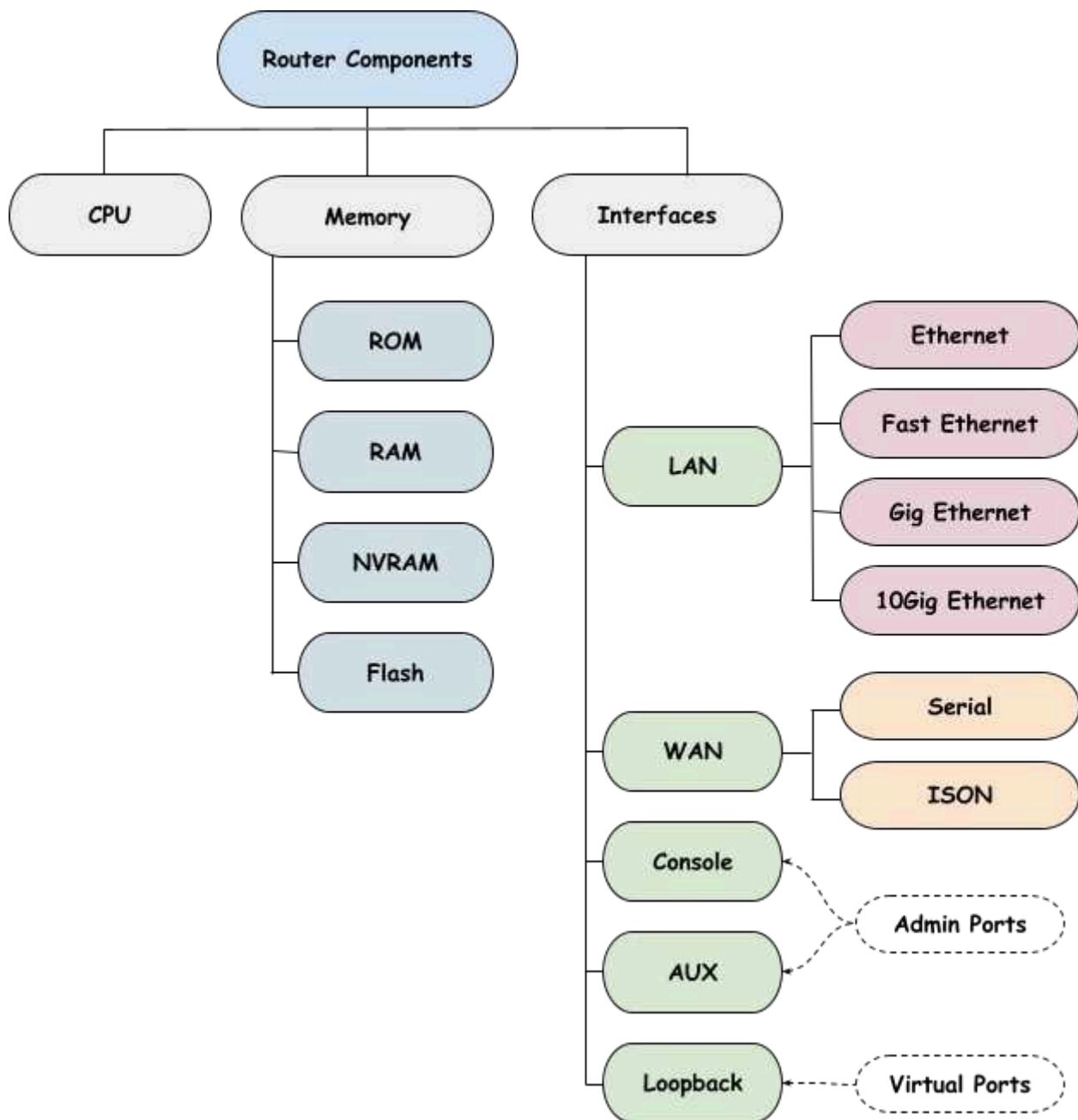
```
R1(config)# enable secret level 5 123
```

وبالتالي اي مستخدم يقدر ينتقل الى Level 5 عن طريق كتابة الأمر Enable 5 ثم ادخال الـ Password

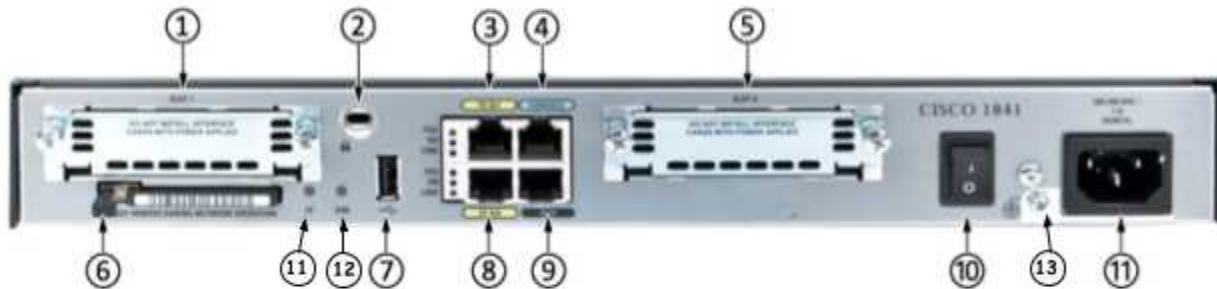
8. Router Components

الراوتر زي اي جهاز Computer او Phone او Tablet يتكون من مجموعة من الأجزاء الرئيسية زي الـ CPU والـ

Operating System و الـ Memories



مثال على مكونات CISCO Router 1841

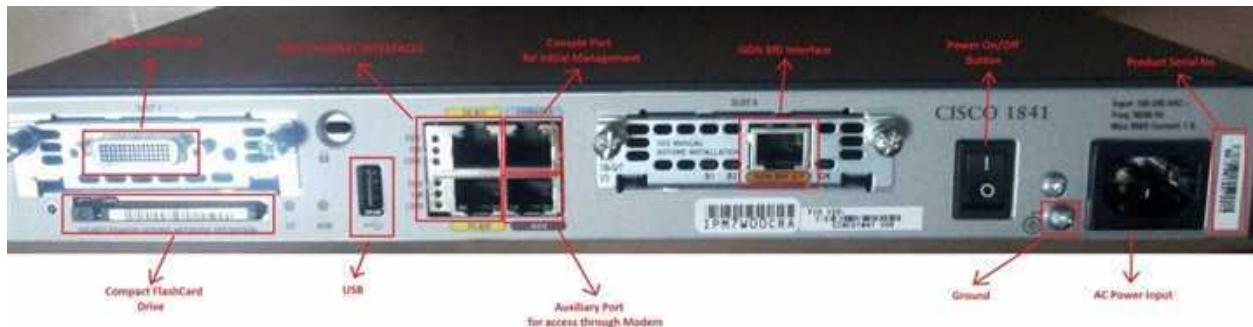


1. Slot 1 (WIC, VWIC—data only, or HWIC)
2. Kensington TM security slot
3. Fast Ethernet ports and LEDs
4. Console port
5. Slot 0 (WIC, VWIC—data only, or HWIC)
6. CompactFlash memory card slot
7. USB port
8. Fast Ethernet ports and LEDs
9. Aux port
10. On/Off switch
11. Input power connection
12. AIM LED
13. Chassis ground connection
14. CompactFlash (CF) LED

و Slot 1 و Slot 0 دی اماكن لاضافة وحدات او Modules اضافية زي WIC, VWIC, HWIC .. وكل نوع من ال WICs دی بيبقى فيها Interfaces من نوع معين .. يعني مثلا لو عايز ازود WAN Interfaces ممكن اشتري WIC-2T Module بيبقى فيه اتنين Serial Ports واركه في أحد ال Slots الموجودة، ودا شكل ال .WIC-2T

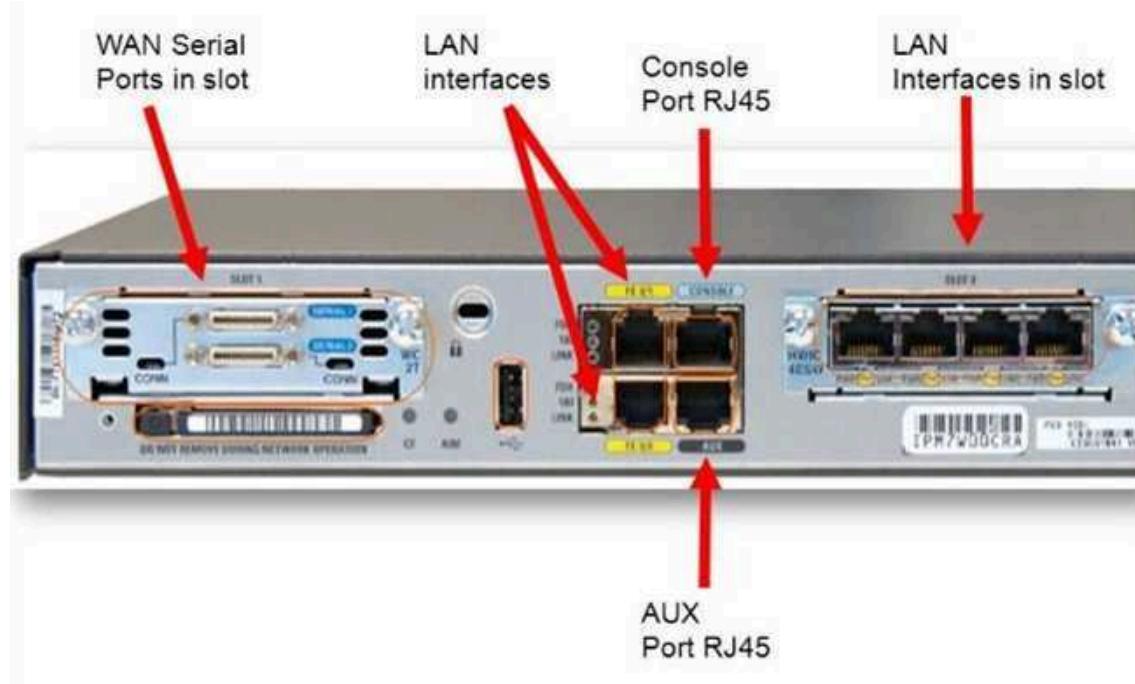


صورة اوضح لنفس الراوتر بعد تركيب Modules hqhtdm



8.1. Router Interfaces

الراوتر يتكون من مجموعة من الـ **Interfaces** .. ويمكن تقسيمها الى ثلاثة



• الـ **Console** زى الـ **Admin Ports** اللي بيتوصل بالـ PC عشان نعمل الـ **Configurations** الأولية للـ

SSH او في حالة لو في مشكلة خدمة الـ **Telnet** او الـ **Router**

الـ **AUX Port** كان بيستخدم برضو لعمل الـ **Configuration** باستخدام Router

Fax Card و

• ال LAN Interfaces زي Ethernet و Fast Ethernet و Giga Ethernet

Gig Ethernet هو ال Standard Protocol المستخدم داخل ال LAN يعني Ethernet. Ethernet Interface مبقاش يستخدم دلوقتي لان Fast سرعته 10Mbps و دي سرعة بطئه مقارنة بالسرعات الموجودة حاليا زي 100Mbps لـ Gig Ethernet و 1Gbps لـ Ethernet.

• ال WAN Interfaces هي ال المستخدمة لربط ال Router بال Internet او ال WAN

Zي ال Serial Interface المستخدم .. وشكل ال مختلف على حسب نوع ال WAN. Serial Interface بيعتبر بحسب نوع ال DSL تستخدم Frame Relay وال Leaseline Technology .. اما ال RJ11 Interface في الرواتر، بنستخدم تحويلة RJ11 ولو مفي RJ11 Interface في الرواتر، بنستخدم تحويلة RJ11.

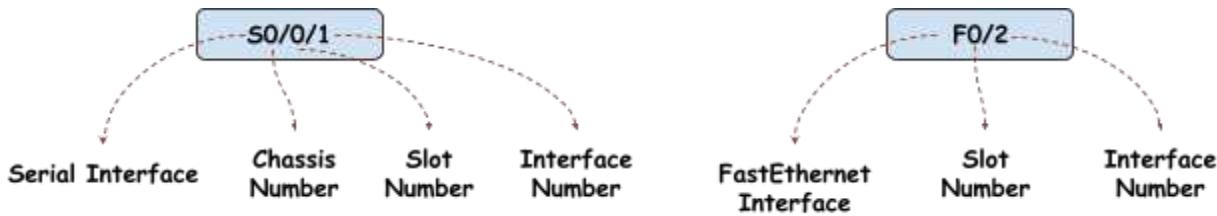
• GBIC Module أو GBIC Interface

Gigabit Interface Converter هو نوع من الوحدات القابلة للتبديل (Hot-swappable) تُستخدم في أجهزة الشبكة، زي **الرواترات والسويتشارتس** بسرعات 1 جيجابت في الثانية (Gbps)، وفي منه Modules بندعم كابلات الألياف الضوئية .. وفي بيدعم كابلات النحاس RJ45.

• SFP Small Form-Factor Pluggable يقدم نفس وظيفة ال GBIC لكن بحجم أصغر وكفاءة أعلى. وبيدعم سرعات أعلى.



كل ياخد رقم زي مثلاً 50/0 او 50/0/1 أو F0/0 أو F0/2



8.2. Router Memories

- الـ **RAM** بتسخدم لتخزين البيانات والعمليات اللي بيقوم فيها الراوتر في الوقت الفعلي. زي الـ جداول التوجيهية (Routing Tables)، التوصيلات (Connections)، وأي بيانات بتتغير بشكل مستمر. واي **Configurations** بتخزن في ملف اسمه **running-config** .. والـ **Configurations** بتعملها .. والـ **RAM** .. واول ما الراوتر يطفي او يعمل **Reload** الإعدادات دي بتطير.
- الـ **ROM** بتحتفظ بالبيانات بشكل دائم (عكس الـ **RAM**). وبيكون فيها الـ **POST Program** والـ **bootstrap loader**، اللي هو برنامج التمهيد اللي بيشتغل أول ما الراوتر يشتغل.
- الـ **NVRAM** بتخزن الإعدادات اللي يحتاجها الراوتر علشان يشتغل بشكل صحيح والـ **Configurations** اللي بتعملها **Save**، والإعدادات دي بتخزن في ملف اسمه **.startup config**.
- الـ **Flash Memory** بتسخدم لتخزين نظام التشغيل الخاص بالراوتر او السويفتش (IOS) .. ممكن تعتبرها زي الـ **Hard Disk**. وممكن نعدل فيها ونضيف اكتر من IOS.

8.3. Router Boot Sequence

أول ما يتوصل بالكهرباء أو يتم إعادة تشغيله، يتبدأ عملية تشغيل الأجهزة والدوائر الداخلية داخل الراوتر. في النقطة دي، يبدأ الراوتر في التحضير للبدء في تحميل النظام.

1. المرحلة الأولى: الراوتر يبدأ في تنفيذ Power-On Self Test أو POST Program والتي يتضمن التالي

- إجراء اختبار على الأجهزة الداخلية (الـ Hardware) للتأكد من إنها شغالة عن طريق إرسال Signal ..

لأن عملية القلاع مش هتم لو في بعض الأجزاء مش شغالة او فيها مشكلة (زي الـ RAM والـ CPU).

- الاختبار ده بيشمل الذاكرة (RAM)، المعالج، والمنافذ (Ports)، والأجهزة الداخلية الأخرى.

3. المرحلة الثانية: تحميل الـ Boot Loader

بعد كدا الراوتر يقوم بتحميل الـ Bootloader من الـ ROM.

الـ Boot Loader هو برنامج صغير جداً، وظيفته إنه يبدأ عملية تحميل نظام التشغيل الرئيسي (الـ

IOS في راوترات Cisco) من الـ Flash Memory ويعمل منه Copy ويحطه في الـ RAM

بعد تحميل نظام التشغيل، الراوتر يبدأ في تحميل الـ Configurations الخاصة به من الـ NVRAM

الـ startup-config دي بتبقى موجودة في ملف اسمه Configurations

الراوتر يدور على ملف الـ Startup-config في الـ NVRAM ولو موجود بيحملو في ملف الـ

RAM على الـ Running-config

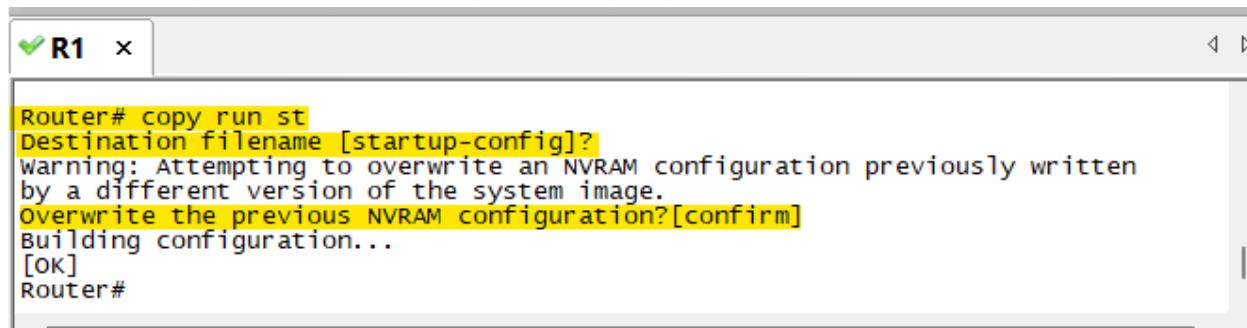
8.4. Save Configurations

لحفظ الاعدادات من الـ Running-config الى الـ Startup-config

```
R1# copy running-config startup-config
```

Shortcut: copy run st

بعد كتابة الامر دا .. الراوتر بيأكده عليك لو عايز تعمل Save على الـ Startup-config



```
R1# copy run st
Destination filename [startup-config]?
Warning: Attempting to overwrite an NVRAM configuration previously written
by a different version of the system image.
Overwrite the previous NVRAM configuration? [confirm]
Building configuration...
[OK]
Router#
```

في امر تاني بيحفظ الـ Configuration في الـ Startup-config مباشرة من غير تأكيد

```
R1# write memory
```

shortcut: wr

لنسخ او تحميل الاعدادات من الـ Startup-config الى الـ Running-config

```
R1# copy startup-config running-config
```

Shortcut: copy st run

العملية دي بتحصل تلقائيا خلال عملية الـ POST .. وبنستخدمها في عملية الـ Password Recovery

لحفظ او تحميل الـ Configurations من والى TFTP Server وهو Server يستخدم لرفع وتوزيل الملفات
Backup الصغيرة وعمل

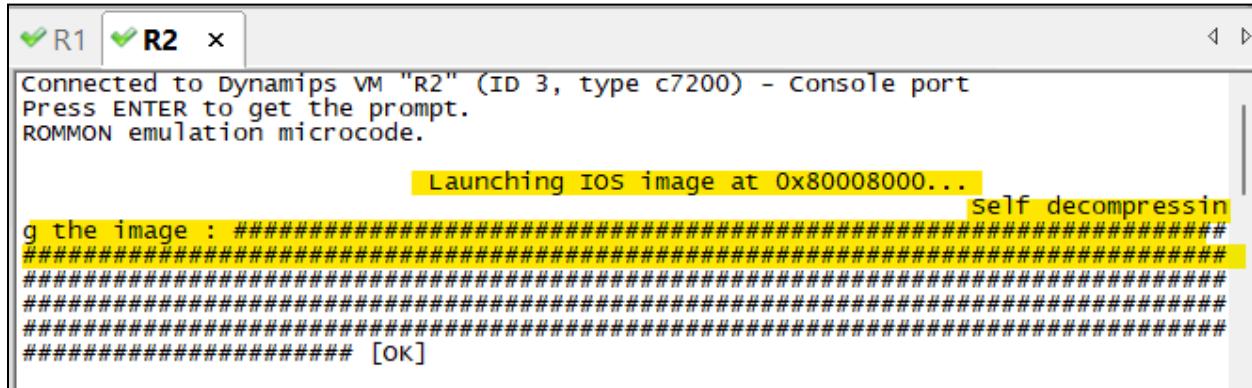
```
R1# copy startup-config tftp
```

```
R1# copy tftp startup-config
```

وممكن تستخدم Startup-config بدل Running-config

مع بدء تشغيل الراوتر، هتلاحظ عملية الـ Decompressing للنظام (IOS) لتجهيزه لعمل Copy منه الى الـ

RAM



لحفظ الـ Configuration المحفوظة في ملف Startup-config وارجاع الـ Router حالته الاصلية

```
R1# delete startup-config
```

أو

```
R1# erase startup-config
```

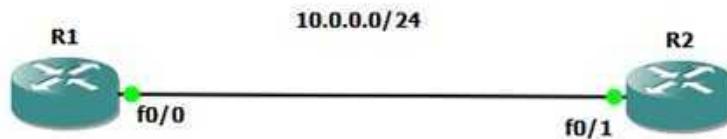
ممكن كل الملفات الموجودة في الـ NVRAM أو ملف معين

```
R2# delete nvram:?
nvram:ifIndex-table    nvram:persistent-data    nvram:private-config
nvram:startup-config   nvram:underlying-config
```

9. Telnet & SSH

9.1. Telnet

Telnet work on port 23 TCP



Set IPs

```
R1(config)# int f0/0
R1(config-if)# ip add 10.0.0.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
```

```
R2(config)# int f0/1
R2(config-if)# ip add 10.0.0.2 255.255.255.0
R2(config-if)# no shutdown
R2(config-if)# exit
```

Open port 23 on R2 with password 123

```
R2(config)# line vty 0 4
R2(config-line)# transport input telnet
R2(config-line)# password 123
R2(config-line)# login
```

- هنا انا سمحت بـ 5 session على الـ Router
- الامر `transport input telnet` يسمح بعملية الـ Telnet وممكن اخليه SSH بس او `all`
- اكتب الاثنين جمب بعض او اكتب `all`
- عملت `Login` ثم `Password` .. ولازم الترتيب هنا يعني لازم اكتب الـ Password الاول وبعددين اكتب `Login`
- عشن اعمل Telnet من R1 على R2 بكتب الـ IP بتاع R1 او R2

لو عندي اكتر من راوتر وبعمل عليهم Configuration وكل شوية اطلع من واحد وادخل على الثاني .. كل ما اطلع وادخل تاني هيطلب مني الـ `Session` من غير انهائها > اضغط على +

`x shift + 6`

عرض الـ Sessions المفتوحة >> اكتب الامر

```
R1# show sessions
Conn Host Address Byte Idle Conn Name
* 1 10.0.0.2 10.0.0.2 0 0 10.0.0.2
```

الرقم المكتوب قبل الـ IP هو رقم الـ Session عشان لو عايز استرجعها وعلامة * معناها ان دي اخر Session كانت مفتوحة ولاسترجاعها بسهولة اضغط Enter

لاسترجاع معينة اكتب أمر `resume` ثم رقم الـ Session او رقم الـ Session ثم `resume`

```
R1# resume 1
```

عرض الـ Users اللي داخلين على الـ Router

```
R2# show users or who
Line User Host(s) Idle Location
* 0 con 0 idle 00:00:00
98 vty 0 idle 00:07:15 10.0.0.1

Interface User Mode Idle Peer Address
```

ممكن امسح معينة عن طريق كتابة

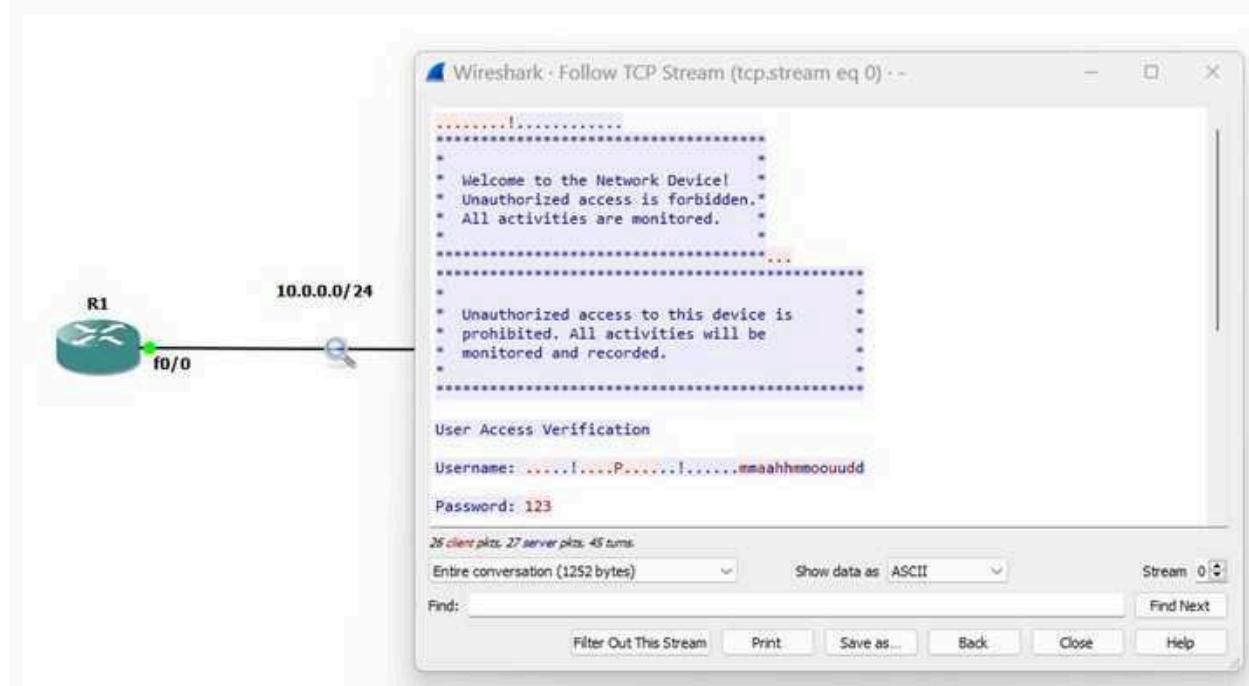
```
R2# clear line 98
[confirm]
[OK]
R2#show users
Line User Host(s) Idle Location
* 0 con 0 idle 00:00:00

Interface User Mode Idle Peer Address
```

عمل Telnet لـ Password و Username

```
R2(config)# line vty 0 4
R2(config-line)# no password
R2(config-line)# no login
R2(config-line)# login local
R2(config-line)# exit
R2(config)# username mahmoud password 123
R2(config)# username mohammed privilege 15 password 123
```

لو عملنا لـ Telnet Traffic عن طريق Wireshark بيفى هنلاقي ان الا Traffic Capture بيرد عليك بنفس الا الا Password Character اي `123` لا الا `mahmoud` او `mohammed` .. وكمان لو في حاجة اتبعتنى من الطرف الثاني مش بيترد عليها



9.2. Secure Shell - SSH

Work on port 22 TCP

9.3. SSH Configuration

Configure the Domain Name

```
R2(config)# ip domain-name example.com
```

Generate RSA Key Pair

```
R2(config)# crypto key generate rsa
```

The name for the keys will be: MyRouter.example.com Choose the size of the key modulus in the range of 360 to 4096 for your General Purpose Keys. Choosing a key modulus greater than 512 may take a few minutes.

How many bits in the modulus [512]: 1024

% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]

```
R2(config)#
```

```
*Mar 1 03:28:07.251: %SSH-5-ENABLED: SSH 1.99 has been enabled
```

- ممكن تستخدم الـ RSA او الـ EC وهي اختصار لـ Elliptic curve ودا احدث وافضل من الـ RSA بس

مش مدعوم على الـ IOS القديمة

- ممكن تحديد حجم التشفير بعد كلمة rsa او اضغط Enter وهو هيقال .. وكل ما زودت الحجم كل ما

كان أمان وكل ما كان الـ Load على الـ Processor اكبر .. وبفضل ان الـ Key لا يقل عن 1024

```
R2(config)# ip ssh version 2
```

الامر دا لتفعيل SSHv2

```

R2# show crypto key mypubkey rsa
% Key pair was generated at: 03:28:07 UTC Mar 1 2002
Key name: R2.example.com
Storage Device: not specified
Usage: General Purpose Key
Key is not exportable.
Key Data:
30819F30 0D06092A 864886F7 0D010101 05000381 8D003081 89028181 00A6D815
F39CE137 57ED7EB7 51B953A0 DB59AB9A A3AB92FF F13A745C 45086D9A 4E0257D5
63F7C8C2 44CDBBD5 7F26984D 216779CE 9434DC07 202615AD 5B69986A 5F726867
8F61CDB2 B5B36495 121CB52D D35B6867 DECF29E3 CD6DCCE0 B19A64E1 609FB719
15BB9BE2 E8DE59F6 EEA198D1 144B3AB4 44B00A71 722674D3 D52FF50F A9020301 0001
% Key pair was generated at: 03:28:07 UTC Mar 1 2002
Key name: R2.example.com.server
Temporary key
Usage: Encryption Key
Key is not exportable.
Key Data:
307C300D 06092A86 4886F70D 01010105 00036B00 30680261 00BEA165 C7414888
7F4AD2B1 70169CB0 3296FF5E F963886B 1054221A 4BF38955 B5C80838 A81CB2CC
909B19C5 C1141203 B3987D0C DCA5EAE5 665789B5 DF594B0D 22D8B34F 42BFD47C
BF1DFAB5 A78C324B C6872A60 78E8CB7B 8ED63A56 48D1ACB0 05020301 0001

```

عرض الـ **Key**

```

R1# ssh -l mahmoud 10.0.0.2
Password:
R2>

```

عمل SSH على الراوتر

```

R2# show ip ssh
SSH Enabled - version 2.0
Authentication timeout: 120 secs; Authentication retries: 3

```

عرض بعض المعلومات

```

R2# show ssh
Connection Version Mode Encryption Hmac State Username
98      1.99    IN    aes128-cbc hmac-sha1 Session started mahmoud
98      1.99    OUT   aes128-cbc hmac-sha1 Session started mahmoud
%No SSHv1 server connections running.

```

10. Banner

الـ Banner عبارة عن رسالة دعائية بتظاهر اول ما تعمل Connect على الـ Router سواء بالـ Console او الـ SSH او الـ Telnet وفي منها اكتر من نوع

- MOTD → Message of the day banner

هي رسالة ترحيبية

- Login Banner

رسالة ترحيبية بتظاهر لو عملت Connect على الراوتر بالـ Telnet او SSH اما في حالة الـ Console او الـ SSH

لازم اكون عامل Password

- exec banner

بتظاهر بعد عملية تسجيل الدخول

10.1. MOTD

```
R1(config)# banner motd ?
LINE c banner-text c, where 'c' is a delimiting character

R1(config)#banner motd %
Enter TEXT message. End with the character '%'.
=====
Mahmoud Tarek
=====

2024 %
```

10.2. Login Banner

```
R1(config)# banner login ?
LINE c banner-text c, where 'c' is a delimiting character

R1(config)#banner motd %
Enter TEXT message. End with the character '%'.
=====
Login and die
=====%
```

11. Filtration

بدل ما تستخدم Show run وبعدين تدور فال Configuration كلها « ممكن تستخدم طرق لعرض الجزء اللي
محتاجه فقط

مثلا كتابة

```
R1# show running-config int f0/1
Building configuration...

Current configuration : 83 bytes
!
interface FastEthernet0/1
    no ip address
    shutdown
    duplex auto
    speed auto
end
```

ممكن تستخدم علامة ال PIPE ودي بتطبق Filter على الامر اللي كتبته .. مثلا لعرض Section معين من ال Configuration

```
R1# show running-config | ?
append      Append redirected output to URL (URLs supporting append
operation
            only)
begin      Begin with the line that matches
exclude    Exclude lines that match
include    Include lines that match
redirect   Redirect output to URL
section    Filter a section of output
tee        Copy output to URL
```

```
R1# show running-config | section line c
line con 0
exec-timeout 0 0
privilege level 15
logging synchronous
```

ممكن تستخدم `Include` مع الـ PIPE لعرض كل السطور اللي بتحتوي على كلمة معينة

```
R1# show run | include interface
interface FastEthernet0/0
interface FastEthernet0/1
```

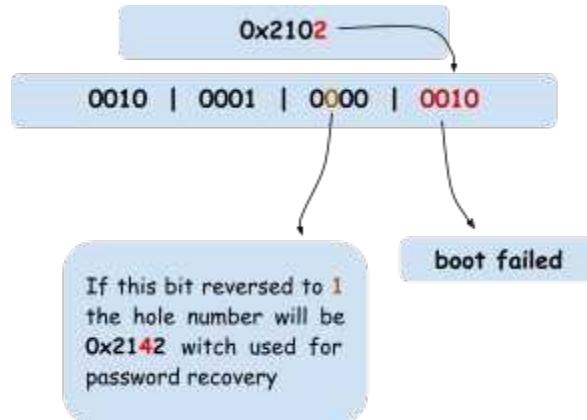
- ممكن تستخدم `Exclude` مع الـ PIPE لعرض كل السطور ما عدا اللي بتحتوي على كلمة معينة
- او تستخدم `begin` لعرض الـ Configuration ابتداءا من كلمة معينة

12. Router Password Recovery

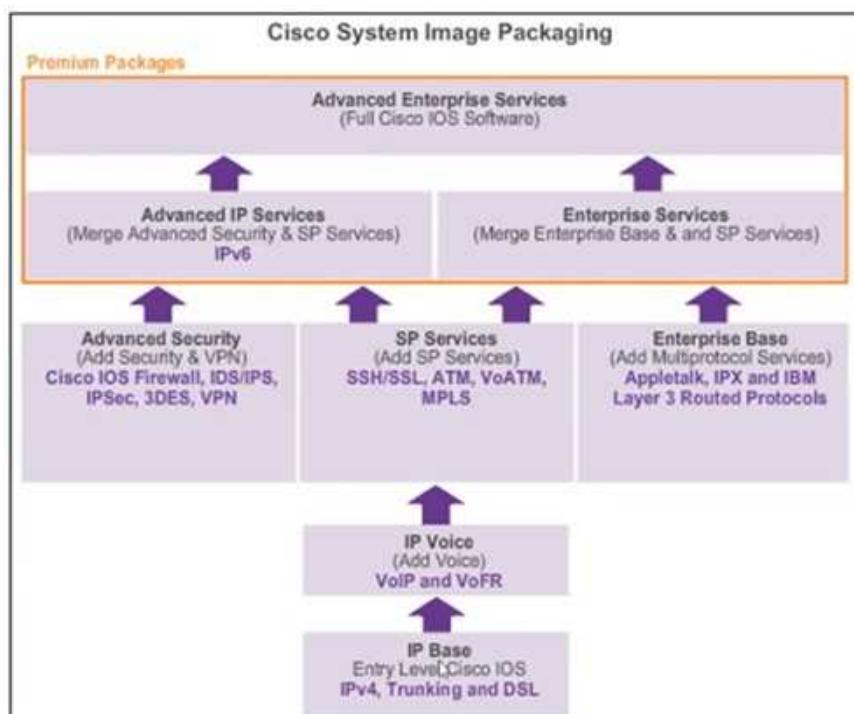
Configuration Register

هي قيمة Configurable موجودة على الـ Router بنظام الـ Save في الـ NVRAM بيعملها بـ Hexadecimal في الـ Register.

وقيمتها 0x2102 افتراضياً.

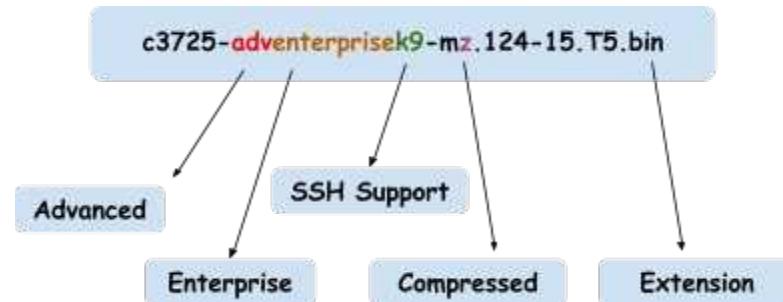


انواع الـ Router Operating Systems اللي اقدر احطها على الـ



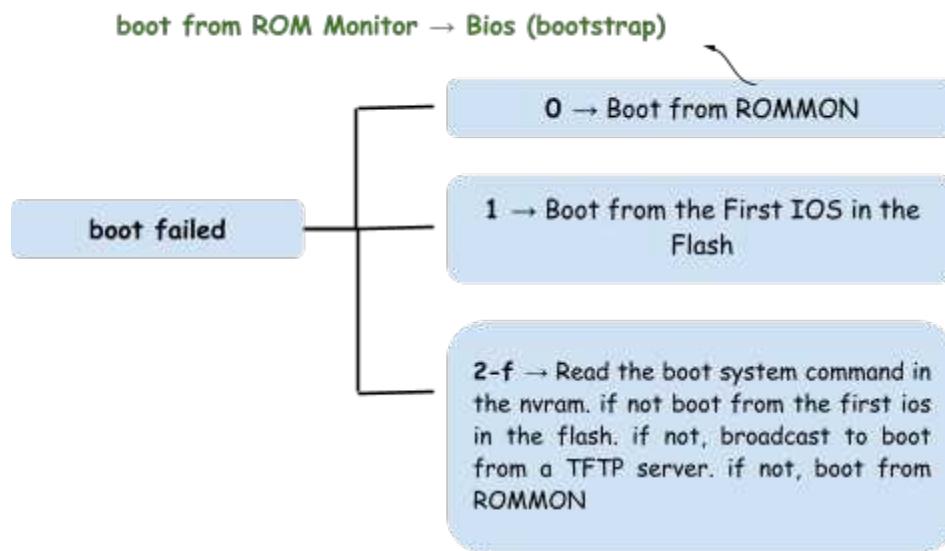
اقل OS هو الـ IP Base ودا بيدعم Features قليلة وكل ما اعلى الـ OS كل ما الـ IP Base تزيد

نأخذ مثال لأحد إصدارات الـ Router OS



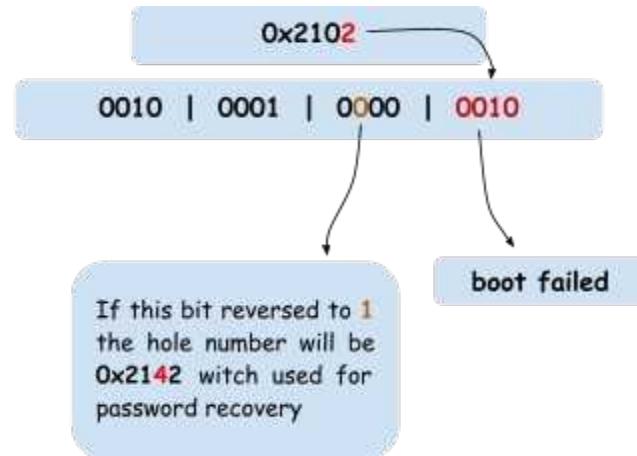
- لو عندي OS معين وشتريت مثل OS اعلى منو .. المفترض امسح الـ OS القديم من الفلاش وبعدين

احط الجديد وممكن لو في مساحة على الفلاش احط الاثنين جنب بعض ويبقى عندي Dual boot



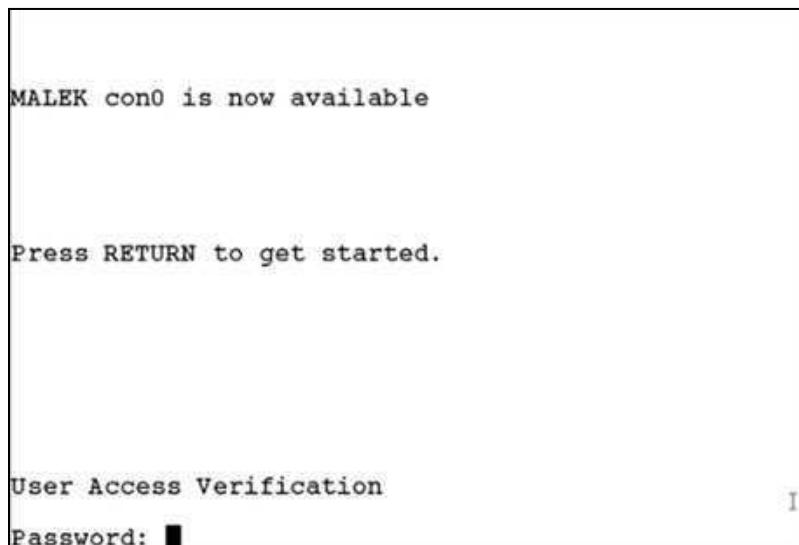
The command saved in the nvram can be:

- Router(config)# boot system flash
- Router(config)# boot system tftp://10.0.0.1/gdghd



التغيير دا مقدرش اعمل الا من خلال الوصول للراوتر بشكل مباشر عن طريق كابل الكونسول

لو عملت Password recovery ونسيتو وعايز اعمل Password recovery



- اول حاجة بطي في ال Router

- بعمل اعادة تشغيل وفي أول 60 ثانية بضغط على ctrl + break او اضغط

على تابة Teeraturm في حالة Send Break ثم Control

```
Readonly ROMMON initialized
rommon 1 >
rommon 1 >
rommon 1 > █
```

```

rommon 2 > confreg

        Configuration Summary
        (Virtual Configuration Register: 0x2102)
enabled are:
load rom after netboot fails
console baud: 9600
boot: image specified by the boot system commands
      or default to: cisco2-c2801

do you wish to change the configuration? y/n [n]:  
  

rommon 3 > confreg 0x2142  
  

You must reset or power cycle for new config to take effect
rommon 4 > reset  
  

System Bootstrap, Version 12.4(13r)T, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 2006 by Cisco Systems, Inc.
PLD version 0x10
GIO ASIC version 0x127
c2801 platform with 131072 Kbytes of main memory
Main memory is configured to 64 bit mode with parity disabled  
  

 Readonly ROMMON initialized
program load complete, entry point: 0x8000f000, size: 0xcb80

```

- الخطوة دي بتعمل Reset لـ NVRAM Bypass يعني هيتخطى عملية الـ Running-config على ملف الـ Startup-config Loading

وهنا لازم اخذ بالي اني لو غيرت الـ Password وعملت « save » هيعمل OverwriteStartup-config الجديدة على الـ Configuration القديمة اللي موجودة في الـ Configuration وبالتالي لازم بعد ما اعمل Copy لـ Pass لـ Reset « اعمل Save Password وبعددين اغير الـ Running-config لـ Startup-config

- ناخد بانا ان قيمة الـ Config Register لسا متغيرتش .. يعني لو عملت reload على الوضع دا هيعمل Reset تاني وبالتالي لازم اغيرها عن طريق الامر:

```
R(config)# config-register 0x2102
R(config)# do wr
```

- في امر مش موجود في الـ Help بتاع الـ Router وهو اللي بيسمح بعمل ROMMON mode من خلال الـ Password Reset

```
R(config)# service password-recovery
```

بس لو عمل Disable للأمر دا مش هقدر اعمل Password Reset لـ Configuration مع استعادة الـ PasswordReset القديمة وعشان ادخل على الـ ROMMON Mode لازم اطبق الخطوات اللي فاتت مع الضغط على

Ctrl + Break مرتين

ctrl + scrlk ctrl + insert ctrl → ctrl + fn

13. CDP - CISCO Discovery Protocol

هو L2 Protocol خاص بشركة Cisco وفي بعض الشركات زي HP معاها License باستخدامه على أجهزتها ..
بس بعد كدا بقى Standard .. يعمل البروتوكول في الـ Data link Layer والهدف منه هو استكشاف الـ Neighbor Devices ما عدا الـ Firewalls .. بروتوكول الـ CDP يبقى Enabled by default على اجهزة Cisco .

طريقة العمل:

- لازم الـ Interface يكون UP
- كل 60 ثانية اجهزة Cisco بتبعث Direct Connected Message للجهاز اللي بيها عشان يعرفهم انه Alive
- لو الجهاز بطل بيعت CDP « جارو هيستنى لمدة 180 ثانية والفترة دي اسمها Hold Time

استخدامات بروتوكول الـ CDP

- يستخدم في الـ Troubleshooting .
مثلا لو في مشكلة في الـ Ping « ممكن تكون المشكلة L2 مش L3 .. وبالتالي ممكن تعرض الـ CDP
لان لو في مشكلة في L2 يبقى كل الـ Layers اللي فوقها مش هتشتغل .

رسم Topology للشبكة.

- من خلال أمر `Show cdp neighbor` على اي Direct Connected Devices تقدر تعرض الـ Topology
وبعدين ممكن تعمل Telnet على جهاز تاني وتشوف نفس الكلام لحد ما توصل لـ Interface كاملة للشبكة .

- بروتوكول الـ CDP يستخدم لتبادل بعض المعلومات بين الاجهزة .
مثلا لو عندي IP Phone مرتبط بالسويفتش .. الـ IP Phone يحتاج IP واحيانا Vlan « فالـ IP Phone بيستخدم بروتوكول CDP لارسال الـ Vlan للـ Switch

- برضو الا IP Phone بيحتاج Power وممكن ياخدو عن طريق تقنية PoE .. بس كل IP Volt ممكن يحتاج Power مختلف وهنا دور الا CDP انه بينقل معلومة الجهاز تحتاج Phone كام.

احتياطات

- بروتوكول CDP بينقل معلومات كتير جدا زي ما عرفنا وبالتالي يفضل نعمله Disable على الا Interfaces اللي مرتبطة بال WAN عشان ميحصلش Attack علية وحد يلقط التрафيك دا.
- يفضل نعمل Disable للـ CDP على الا PCs المرتبطة بـ Interfaces لأن ممكن المستخدم اللي عليها عن طريق بعض البرامج يبعث CDP ويعمل نفسه جهاز Cisco ويلقط المعلومات دي.

في الاب استخدمت IOS Switch 15.5 و IOU Router 15.1

لعرض معلومات عن بروتوكول CDP

```
R# show cdp
Global CDP information
  Sending CDP packet every 60 seconds
  Sending a holdtime value of 180 seconds
  Sending CDPv2 advertisements is enabled
```

لعرض معلومات عن الاجهزة المتصلة

```
R# show cdp neighbor
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater
```

Device ID	Local Interface	Holdtime	Capability	Platform	Port ID
SW1	Eth 0/0	135	R S I	Linux uni	Eth 0/1
R2	Eth 0/1	161	R B	Linux uni	Eth 0/0

Device ID	اسم الـ Neighbor Device
Local Interface	البورت من جهة
Capability	نوع الجهاز المقابل .. في سوبيتشات ممكن تشتفل كراوتر
Platform	نوع النظام .. مثل IOS on UNIX هي اختصار ل IOU
Port ID	البورت من جهة الجهاز اللي متصل بيها
Hold time	عبارة عن عدد تنازلي من 180 ثانية .. يفضل يقل لحد ما توصلني CDP وبعدين يبدا من 180 ثاني

عرض تفاصيل اكتر عن الأجهزة المتصلة بالراوتر باستخدام CDP

```
R1# show cdp neighbors detail

-----
Device ID: R2
Entry address(es):
Platform: Cisco 3725, Capabilities: Router Switch IGMP
Interface: FastEthernet0/1, Port ID (outgoing port): FastEthernet0/0
Holdtime : 129 sec

Version :
Cisco IOS Software, 3700 Software (C3725-ADVENTERPRISEK9-M), Version 12.4(15)T5,
RELEASE SOFTWARE (fc4)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Wed 30-Apr-08 18:27 by prod_rel_team

advertisement version: 2
VTP Management Domain: ''
Duplex: half

-----
Device ID: IOU1
Entry address(es):
Platform: Linux Unix, Capabilities: Router Switch IGMP
Interface: FastEthernet0/0, Port ID (outgoing port): Ethernet0/0
Holdtime : 126 sec

Version :
Cisco IOS Software, Linux Software (I86BI_LINUXL2-ADVENTERPRISEK9-M), Version
15.2(CML_NIGHTLY_20180510)FLO_DSGS7, EARLY DEPLOYMENT DEVELOPMENT BUILD, synced to
V152_6_0_81_E
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2018 by Cisco Systems, Inc.
Compiled Thu 10-May-18 02:45 by mmem

advertisement version: 2
VTP Management Domain: ''
Native VLAN: 1
Duplex: half
```

من ضمن المعلومات اللي الـ CDP بيظهرها برضو هي الـ IP لو موجود

أوامر أخرى

- لعمل Disable في الـ Configuration mode على الجهاز كله عن طريق الامر

```
R(config)# no cdp run
```

- لحذف معلومات الـ CDP الحالية عن طريق الامر

```
R# clear cdp table
```

- لعمل Disable لبروتوكول CDP على Interface معين عن طريق الامر

```
R(config)# int e0/0
R(config-if)# no cdp enable
```

- تغيير الـ Timers

```
R1(config)# cdp ?
  advertise-v2      CDP sends version-2 advertisements
  holdtime          Specify the holdtime (in sec) to be sent in packets
  log               Log messages generated by CDP
  run               Enable CDP
  source-interface   Insert the interface's IP in all CDP packets
  timer             Specify rate (in sec) at which CDP packets are sent
```

13.1. LLDP - Link Layer Discovery Protocol

بروتوكول شبيه بالـ CDP لكن Open Standard .. الـ Details اكتر .. ويكون Enabled by default

Hold time 120 ثانية و 30 ثانية Timers

- لتفعيل الـ LLDP

```
R(config)# lldp run
```

- عرض معلومات عن LLDP

```
R# show lldp
```

مشكلة امر **show lldp** انه بيعرض الـ Timer بشكل ثابت .. فلازم استخدم الأمر التالي لعرض الـ Realtime Timers في الـ

```
R# show lldp neighbor details
```

- لتشغيل LLDP للاستقبال فقط او الارسال فقط او الاثنين تحت Interface معين

```
R(config-if)# lldp transmit or lldp receive
```

المعلومات اللي الـ LLDP بينقلها اسمها TLV .. وفي ميزة هنا انى اقدر اخليه ميعتنش بعض المعلومات للـ Neighbors .. من المعلومات اللي اقدر امنعها:

```
IOU1(config)# lldp tlv-select ?  
4-wire-power-management Cisco 4-wire Power via MDI TLV  
mac-phy-cfg IEEE 802.3 MAC/Phy Configuration/status TLV  
management-address Management Address TLV  
port-description Port Description TLV  
port-vlan Port VLAN ID TLV  
power-management IEEE 802.3 DTE Power via MDI TLV  
system-capabilities System Capabilities TLV  
system-description System Description TLV
```

ممكن اكتب no قبل الامر لمنع ارسال المعلومة المحددة

14. Routing

الـ Routing هو عملية توجيه البيانات بين الشبكات المختلفة باستخدام أجهزة التوجيه (Routers). الراوتر يحدد أفضل مسار لنقل البيانات بناء على بروتوكول التوجيه المستخدم، والهدف الرئيسي هو ضمان وصول البيانات بسرعة وكفاءة من المصدر Source إلى الوجهة Destination.

في نوعين من الـ Routing

1. Static Routing

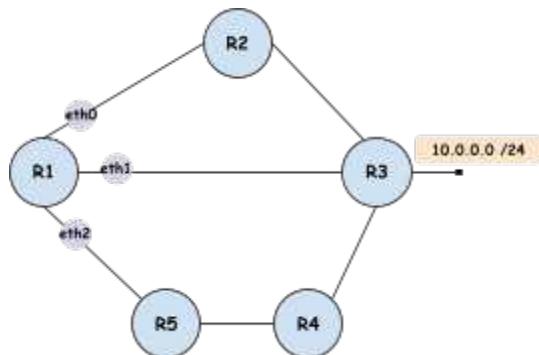
هنا يتم تعريف المسارات يدويا .. والنوع ده بيكون بسيط ومناسب للشبكات الصغيرة أو المسارات الثابتة، لكنه يتطلب تدخل يدوي لو حصل أي تغيير في الشبكة.

2. Dynamic Routing

في النوع ده، الراوترات بتتعلم المسارات تلقائيا باستخدام بروتوكولات زي OSPF أو EIGRP. النوع ده مناسب للشبكات الكبيرة اللي بتتغير بشكل مستمر، ويبوفر مرونة وكفاءة أعلى مقارنة بـ Static Routing.

14.1. Static Routing

لو عندنا الـ Topology الموضح بالصورة:



- في 3 مسارات عشان R1 يصل لـ R3 .. وعشان R3 يختار أفضل مسار، بيبيني جدول توجيه داخل الـ Routing Table اسمه RAM

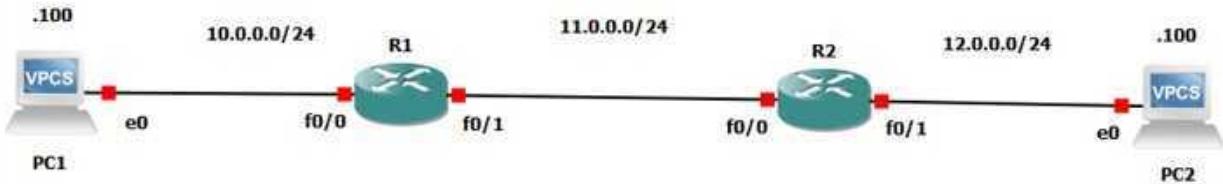
- الـ **Routing Table** بيعمل **Mapping** بين عنوان الشبكة والـ **Exit Interface** اللي هيوصلني للشبكة.

Routing Table

Network	Exit Interface
10.0.0.0 /24	eth 1
...	...

- وبالتالي عشان R1 يصل لشبكة 10.0.0.0/24 <> هيطلع من Interface Ethernet 2
- زي ما قلنا في الـ **Static Routing** يتم تعريف المسارات يدويا .. يعني الـ Admin او المسؤول هو اللي بيبني جدول التوجيه وبالتالي Load Processor أقل على الـ Processor بتابع الراوتر .. بس المشكلة ان المسار اللي اخترته بشكل يدوي ممكن يكون افضل مسار، او ممكن يحصل فيه مشكلة وبالتالي لازم يتم تعريف مسار تاني بشكل يدوي .. بعكس الـ **Dynamic Routing** اللي بيستخدم بروتوكولات بختار افضل مسار لكل شبكة بناءا على خوارزميات معينة، ولو حصل مشكلة في المسار دا، الخوارزمية بتشغل تاني عشان تختار افضل مسار موجود.

14.1.1. Static Routing Lab



- من وظائف الـ Network انها End to End Delivery يعني بتوصل جهازين من شبكات مختلفة بعض.

- هدف الـ Lab دي هي اني اخلي الشبكة Pingable .. يعني لو عملت Ping من PC1 على PC2 يطلع وبكدا اكون بنية تحتية للشبكة (Network Infrastructure).
- الـ IP الخاص بـ Int f0/0 على R1 هو الـ Default Gateway لـ PC1
- بمجرد وضع IP لـ لـ Routing Table وعمل Shutdown Interfaces في جزء من الـ Routing Table يتبني تلقائيا .. وهو الجزء الخاص بالشبكات المتصلة مباشرة بالراوتر (Direct Connected).
- مثلا R2 و PC1 متوصلين مباشرة بـ R1 وبالتالي يتم اضافة المعلومات الخاصة بهم في الـ Routing Table

R1 Routing Table		R2 Routing Table	
Network	Exit Interface	Network	Exit Interface
10.0.0.0/24	C - f0/0	11.0.0.0/24	C - f0/0
11.0.0.0/24	C - f0/1	12.0.0.0/24	C - f0/1

- دلوقتي R1 ناقصه شبكة 12.0.0.0/24 و R2 ناقصه شبكة 10.0.0.0/24 .. ودول اللي هضيفهم بطريقة Static
- لو حاولت تعمل Ping من PC1 على PC2 اللي في شبكة 12.0.0.0/24
- الـ Routing Packet هتروج لـ R1 .. وبعدين R1 هيدور على شبكة 12.0.0.0/24 في الـ Table .. لو مش موجودة هيبعدت لـ PC1
 - "Reply from 10.0.0.1: Destination Unreachable"

○ لكن بعد اضافة شبكة 12.0.0.0/24 في الـ Routing Table الخاص بـ R1 لا يوجه الـ

PC2 لـ Packets

○ عشان يرد بـ PC2 على Reply علـي PC1 <> الـ R2 .. و R2 مش هيعرف يوجه الـ

Reply لـ شبكة 10.0.0.0/24 لأنها مش موجودة عنده .. وبالتالي R1 هيفضل مستني Reply

وبعد وقت معين هيبيت لـ PC1

"Request time out"

اضافة الـ IP's

```
R1# conf t
R1(config)# int f0/0
R1(config-if)# ip add 10.0.0.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# int f0/1
R1(config-if)# ip add 11.0.0.1 255.255.255.0
R1(config-if)# no sh
```

```
R2# conf t
R2(config)# int f0/0
R2(config-if)# ip add 11.0.0.2 255.255.255.0
R2(config-if)# no shutdown
R2(config-if)# int f0/1
R2(config-if)# ip add 12.0.0.1 255.255.255.0
R2(config-if)# no sh
```

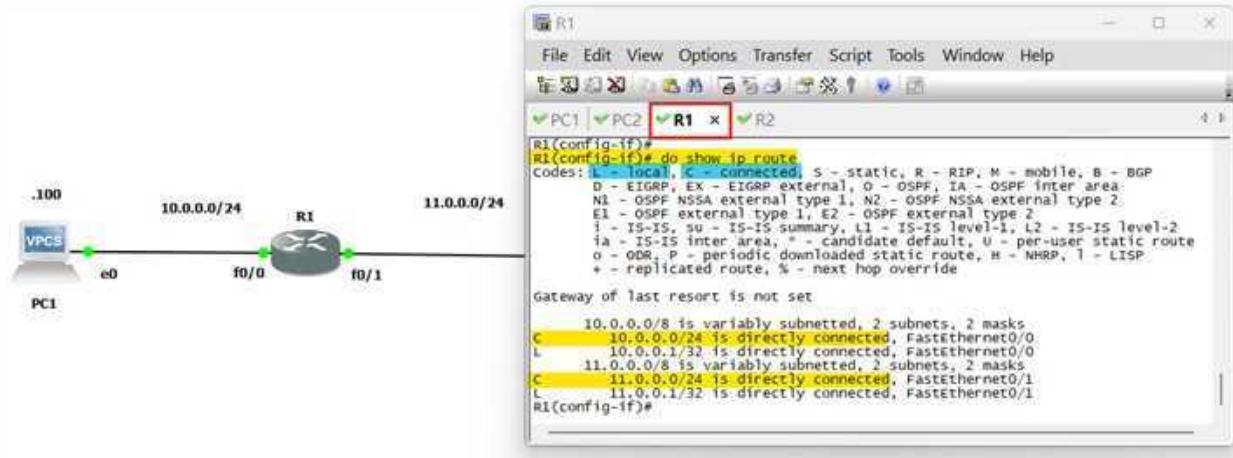
لـ اضافة Virtual PC لـ IP

IP	Subnet Mask	Default Gateway
----	-------------	-----------------

```
PC1> ip 10.0.0.2 255.255.255.0 10.0.0.1
Checking for duplicate address...
PC1 : 10.0.0.2 255.255.255.0 gateway 10.0.0.1
```

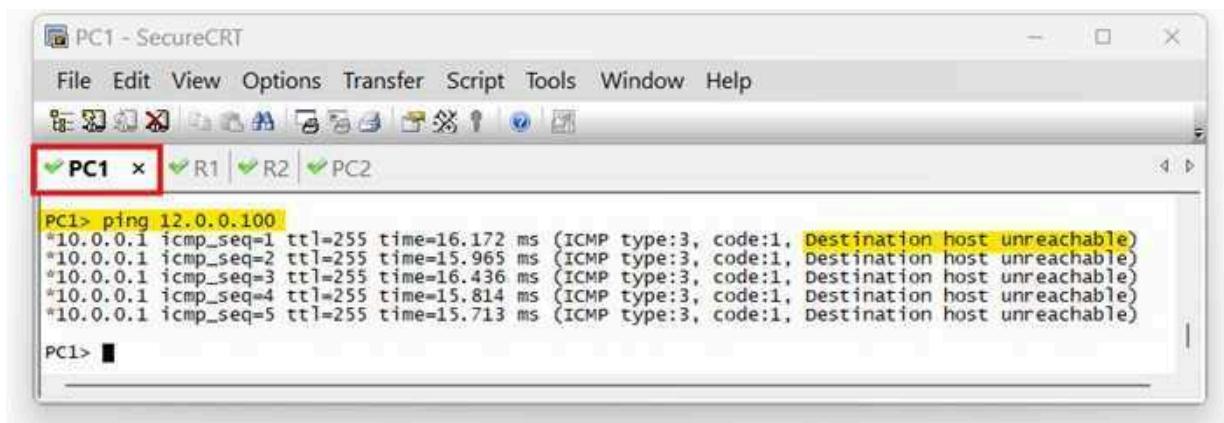
```
PC2> ip 12.0.0.2 255.255.255.0 12.0.0.1
Checking for duplicate address...
PC1 : 12.0.0.2 255.255.255.0 gateway 12.0.0.1
```

عرض الـ Routing Table



قبل اضافة الشبكة بطريقة Static << لو حاولنا نعمل Ping من PC1 على PC2 هنلاقي ان الـ

Unreachable



لاضافة شبكة بطريقة Static

```
R1(config)# ip route 12.0.0.0 255.255.255.0 <exit interface> or <next hop ip>
```

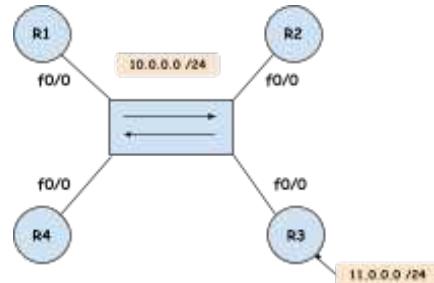
معنى الأمر دا اني بعرف الـ Router لو عايز توصل لشبكة 12.0.0.0 اللي الـ Subnet Mask بتاعها 255.255.255.0 << ممكن توصلها عن طريق الـ Exit Interface اللي هو f0/1 او عن طريق الـ IP الخاص بالـ .11.0.0.2 وهو Next Hope

امتنى نستخدم الـ Next Hop او الـ Exit Interface

لو الراوتر هيعمل توجيه لـ Packets، اول حاجة بيشوف لو في الـ Match في الـ Routing Table .Destination IP

- لو مفيش أي ماتشن (يعني مفيش Default Route أو Route)، الراوتر هي عمل Drop للباكت.
 - لو فيه ماتشن (سواء من Static/Dynamic Route أو حتى Default Route)، الراوتر هيبدأ في تحديد الـ Exit Interface
 - في شبكات الـ Multipoint Link أو الـ Ethernet •
 - لو الـ IP Address متعدد كـ Nexthop
- الراوتر هي عمل Local Lookup في الشبكات المتوصلة مباشرة (Direct Connected) عشان يحدد الـ Exit Interface بالـ IP المرتبط بالـ Nexthop

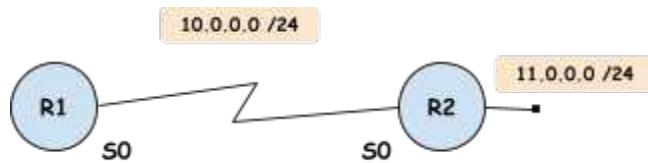
وبعدين هيبيعث ARP Request عشان يعرف الـ MAC بتاع الـ Next Hop وبعدين هيبيعث الـ Layer 2 Frame بعد تغليفها في Packet



- لو الـ IP متعدد كـ Nexthop هيبدأ في توجيه الـ Packets
- الراوتر هيعتبر إن الـ Dest Host متصل مباشرة على الـ Exit Interface، وهيبيعث ARP Request عشان يعرف الـ MAC .. وبما ان الـ Dest Host مش Direct Connected (يعني موجود في شبكة مختلفة)، الـ ARP Request هيفشل، وبالتالي الراوتر مش هيقدر يصل الباكت، وهيحصل Reachability Failure

- لو الـ **Serial Interface** يعني متوصّل بـ **Point-to-Point Link** **Exit Interface**

الراوتر بيبقى متوقع إن فيه جهاز واحد متوصّل باللينك، وكل الـ **Packets** بتتّبع لـ **Exit Interface** اللي بعده .. عشان كده، ممكن تحدّد الـ **IP** أو **Exit Interface** كـ **Nexthop**، والاتنين هيستغلوا بدون مشاكل بس استخدام الـ **Routing** مع الـ **Point-to-point** هيخلّي الـ **Router** بيصل على الـ **Next Hop** أول مرّه عشان يحدّد الـ **Nexthop** اللي هيوصلني بشبكة معينة .. وبعدين يعمل **Reverse Table Lookup** عشان يحدّد الـ **Exit Interface** المرتّب بالـ **Exit Interface** دا



الخلاصة

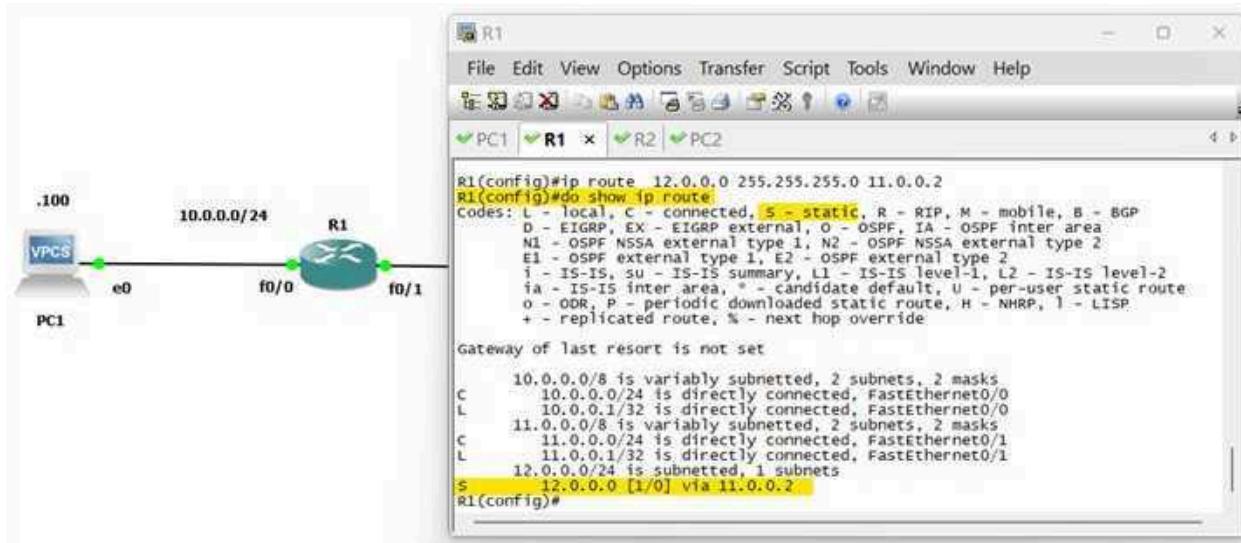
- في شبّكات الـ **Multi-Point** أو الـ **Eaathernet** بـ **يستخدم الـ IP Nexthop**
- أما في شبّكات الـ **Point-to-Point** أو الـ **Serial** اللي بين جهازين فقط >> **يستخدم الـ Exit Interface**
- لو عملت **Default Route** بطريقة الـ **Exit Interface** >> هيطّلع تحذير ان الشبّكة **End-to-End** ودا **Performance** هياّثر على الـ
- لعرض الـ **ARP Table** أو الـ **ARP Cache** .. استخدم الأمر **show arp** .. ودا بيعرض الشبّكات المتوصّلة بالـ **MAC Router** والـ **MAC** الخاص بالـ **Nexthop**

وبالتالي هنضيف الشبكة من خلال الا Next Hop IP Address

```
R1(config)# ip route 12.0.0.0 255.255.255.0 11.0.0.2
```

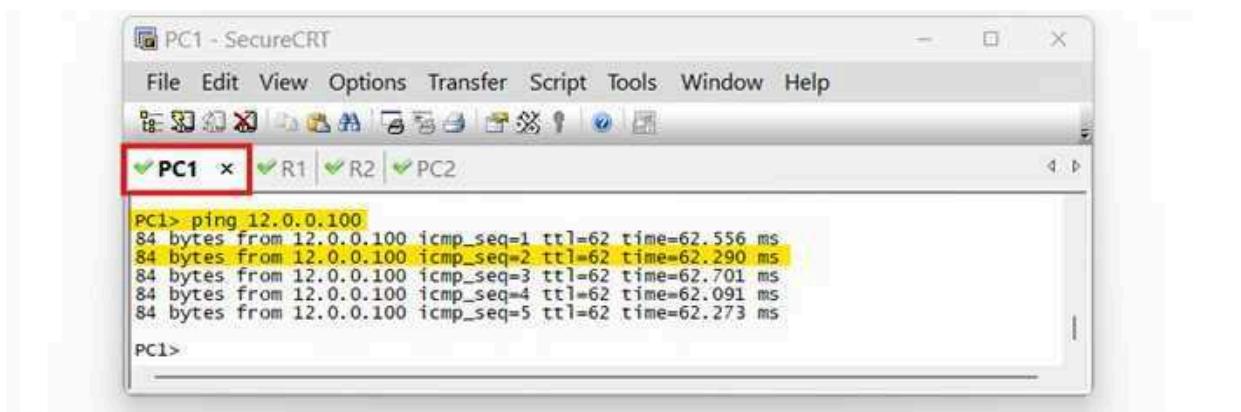
```
R2(config)# ip route 10.0.0.0 255.255.255.0 11.0.0.1
```

لو عرضنا الا Routing Table على R1



هنا لاحظ وجود [1/0] بجانب الا Static Route .. والرقم مشرح في 14.2.2

دلوقيتي لو عملنا Ping تاني



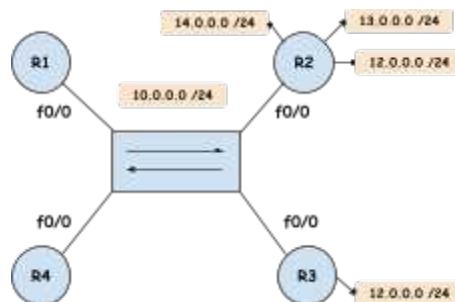
14.1.1.1. Default Routing

الـ Default Route هو مسار افتراضي يبخدمه الراوتر لما ميلقيش مسار محدد في الـ Routing Table للـ Router يبيكع عنده مساحة محددة بتتحمل عدد محدد من الـ Routes او الـ Rows . الـ Destination IP Packets اللي الـ Default Route رايجه له.

طب ليه بستخدم الـ Default Route

- بدل ما اكتب Static Route لكل شبكة بعيدة .. بستخدم الـ Default Route
- الـ Router بيكون عنده مساحة محددة بتتحمل عدد محدد من الـ Routes او الـ Rows
- الخلاصة ان الـ Default Router يروح للـ Default Route لو الشبكة مش عندو في الـ Routing Table

مثلا لو عندي مجموعة من الشبكات على R2 .. بدل ما اعمل Static Routing على كل واحدة منهم <> ممكن اضيف R2 كـ Default Route على R1 مثلا .. وبالتالي لو R1 عايز يصل لـ اي شبكة منهم، هيروح لـ R2.



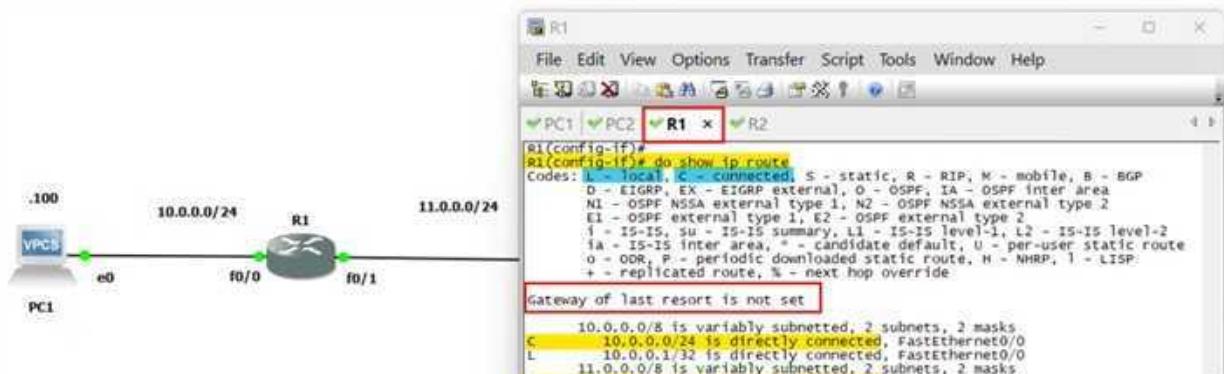
لاضافة Default Route

IP Subnet Mask

```
R1(config)# ip route 0.0.0.0 0.0.0.0 11.0.0.2
```

معنی الأمر اني بخلي الـ Router لو يصل لأي IP هتروج للـ Default Router اللي الـ IP بتاعه 11.0.0.2

لو عرضت الـ Routing Table



Default Router معناها ان مفيش Gateway of last resort •

الـ Routing Table بيظهر بعلامة S* في الـ Default Router •

هيحصل بعض المشاكل لو عملت الـ Default Router Configuration بطريقة خاطئة

يعني لو اضفت R2 كـ Default Router على R1 .. واضفت R1 كـ Default Router على R2 •

لو عملت Ping في الحالة دي من PC1 على IP 20.0.0.1 مش موجود في الشبكة .. مثلا: •

كدا الـ IP مش موجود في الـ Routing Table على R1 .. فهيروح للـ Default Router اللي هو R2 •

وبما انه مش موجود على R2 برضو هيروح للـ Default Router بتاعه اللي هو R1

وكدا هيحصل حاجة اسمها "Routing Loop" •

الحل اني لما بعمل Design للشبكة، لازم احط في الاعتبار ان الـ Default Routing يكون في مسار واحد فقط .. والشبكات الثانية بضيفها بطريقة اخري زي الـ Static مثلًا.

في قيمة في الـ Header بتاع الـ Ping Packet اسمها TTL ودي قيمة بتتمثل في 8 بت .. يعني لها عدد احتمالات $2^8 = 256$.. وفي كل عملية تحويل من Router الى Router الرقم دا بيقل بمقدار واحد .. وبالتالي في حالة الـ Routing Loop الرقم دا هيفضل يقل لحد ما يصل لـ 0 .. وهنالـ Drop هيعمل للـ Packet دي.

عرض الـ Host Routing Table على الـ Windows ممکن تستخدی امر netstat -r او امر

route print

R1(config)# interface loopback <name>					
output omitted ...					
=====					
IPv4 Route Table					
=====					
Active Routes:					
Network Destination	Netmask	Gateway	Interface	Metric	
0.0.0.0	0.0.0.0	192.168.2.1	192.168.2.100	281	
127.0.0.0	255.0.0.0	On-link	127.0.0.1	331	
127.0.0.1	255.255.255.255	On-link	127.0.0.1	331	
127.255.255.255	255.255.255.255	On-link	127.0.0.1	331	
192.168.2.0	255.255.255.0	On-link	192.168.2.100	281	
192.168.2.0	255.255.255.0	On-link	192.168.2.3	311	
192.168.2.3	255.255.255.255	On-link	192.168.2.3	311	
192.168.2.100	255.255.255.255	On-link	192.168.2.100	281	

نلاحظ اول Route يشير الى الـ Default Route على جهازی

والـ Output بيبقى فيه 3 اقسام:

- Interface List - Lists the Media Access Control (MAC) address and assigned interface number of every network-capable interface on the host, including Ethernet, Wi-Fi, and Bluetooth adapters.
- IPv4 Route Table - Lists all known IPv4 routes, including direct connections, local network, and local default routes.
- IPv6 Route Table - Lists all known IPv6 routes, including direct connections, local network, and local default routes.

14.1.1.2. Loopback Interface

في اجهزة CISCO ممكن اضيف Interface وهمي، زي الـ Loopback Interface وله استخدامات كتير منها الـ Testing .. ودائما حاليه بتبقى UP وبيظهر للشبكات الأخرى.

لاضافة Loopback Interface

```
R1(config)# interface loopback <name>
```

امر Show arp بيعرض كل الشبكات المتوصلة بال Router وال MAC Address بتابع كل شبكة او IP

```
R1# show arp
```

Protocol	Address	Age (min)	Hardware Addr	Type	Interface
Internet	10.0.0.1	-	ca01.2b44.0008	ARPA	FastEthernet0/0
Internet	10.0.0.2	17	0050.7966.6800	ARPA	FastEthernet0/0
Internet	10.0.0.100	4	0050.7966.6800	ARPA	FastEthernet0/0
Internet	11.0.0.1	-	ca01.2b44.0006	ARPA	FastEthernet0/1
Internet	11.0.0.2	8	ca02.4db0.0008	ARPA	FastEthernet0/1

14.2. Dynamic Routing

الـ Dynamic Routing عبارة عن Configuration على الـ Router بعمله عشان يستخدمه في توجيه البيانات من الـ Source إلى الـ Destination و اختيار افضل مسار لها .. والـ Software دا عبارة عن Routing Protocol. وفي مصطلحين مهمين هنا:

- بروتوكول موجه - بفتح الجيم- (Routed Protocol) •

هو البروتوكول اللي بيتم توجيئه، زي IPv4 و IPv6 و CLNS و ATP و IPX

- بروتوكول التوجيه او بروتوكول موجه - بكسر الجيم- (Routing Protocol) •

هو البروتوكول اللي بيوجه البيانات وبيختار أفضل مسار لنقلها، يعني مثلا الـ Routing Protocol هو

اللي بيختار المسار توجيه الـ Routed Protocol .. زي OSPF و EIGRP و RIP و IS-IS و ODR و

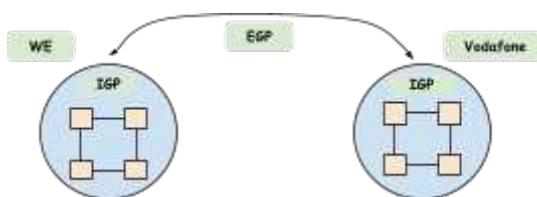
BGP

14.2.1. Types of Dynamic Routing Protocols

في قسمين رئيسيين من بروتوكولات التوجيه، وهم IGP و EGP .. وعشان نفهم الفرق بينهم هنتكلم على مصطلح Autonomous System واللي تعريفه:

"Is a collection of devices under the same administrative authority"

يعني مثلا عندنا شركة WE عبارة عن Autonomous System وعندها مجموعة من الـ Routers متوزعة على أنحاء مصر ومتوصلة بعضها .. والـ Routers بتاعتتها شغالة بـ Routing Protocol معين من نوع IGP .. ومثلا شركة Vodafone والمجلس الأعلى للجامعات دول Autonomous Systems .. كل شركة او AS بيشغل الـ IGP الخاصة بيها بـ Routing Protocol من نوع Routers.



بنربط بين الـ ASes المختلفة بـ Routing Protocol من نوع EGP

Protocol	Interior Gateway Protocols (IGP)					Exterior Gateway Protocols (EGP)
	Distance Vector		Link State		Path Vector	
IPv4	RIPv2	EIGRP	OSPFv2	IS-IS		BGP-4
IPv6	RIPng	EIGRP for IPv6	OSPFv3	IS-IS for IPv6		BGP-MP

IGRP and EIGRP

الـ IGRP والـ EIGRP ملكية خاصة بشركة سيسكو (Cisco Proprietary).

- بروتوكول IGRP بطيء جدا عشان كدا مباقاش موجود في اجهزة Cisco.
- بروتوكول EIGRP يعتبر تحسين (Enhancement) من بروتوكول IGRP.

الـ EIGRP بروتوكول سريع ومحترم جدا كان مملوك لـ CISCO ولكن بعدين بقى

يعني بيشتغل على أي Vendor عادي.

OSPF and IS-IS

بروتوكول OSPF تم تصميمه ليتناسب مع الـ TCP/IP Model عشان كدا هو شائع وأكثر انتشارا من بروتوكول IS-IS اللي تم تصميمه في الأصل ليتناسب مع الـ OSI Model. ولكن IS-IS بروتوكول قوي وسريع جدا عشان كدا تم تطوير اصدار منه اسمه Integrated IS-IS عشان يتناسب مع الـ TCP/IP Model. ومع ذلك Configuration بتاعته لازم يبقى فيها CLNS Address بطريقة ما، ودا اللي مخلي الـ Configuration بتاعته معقدة شوية، وبالتالي الـ OSPF فضل متفوق عليه من ناحية الانتشار والسهولة.

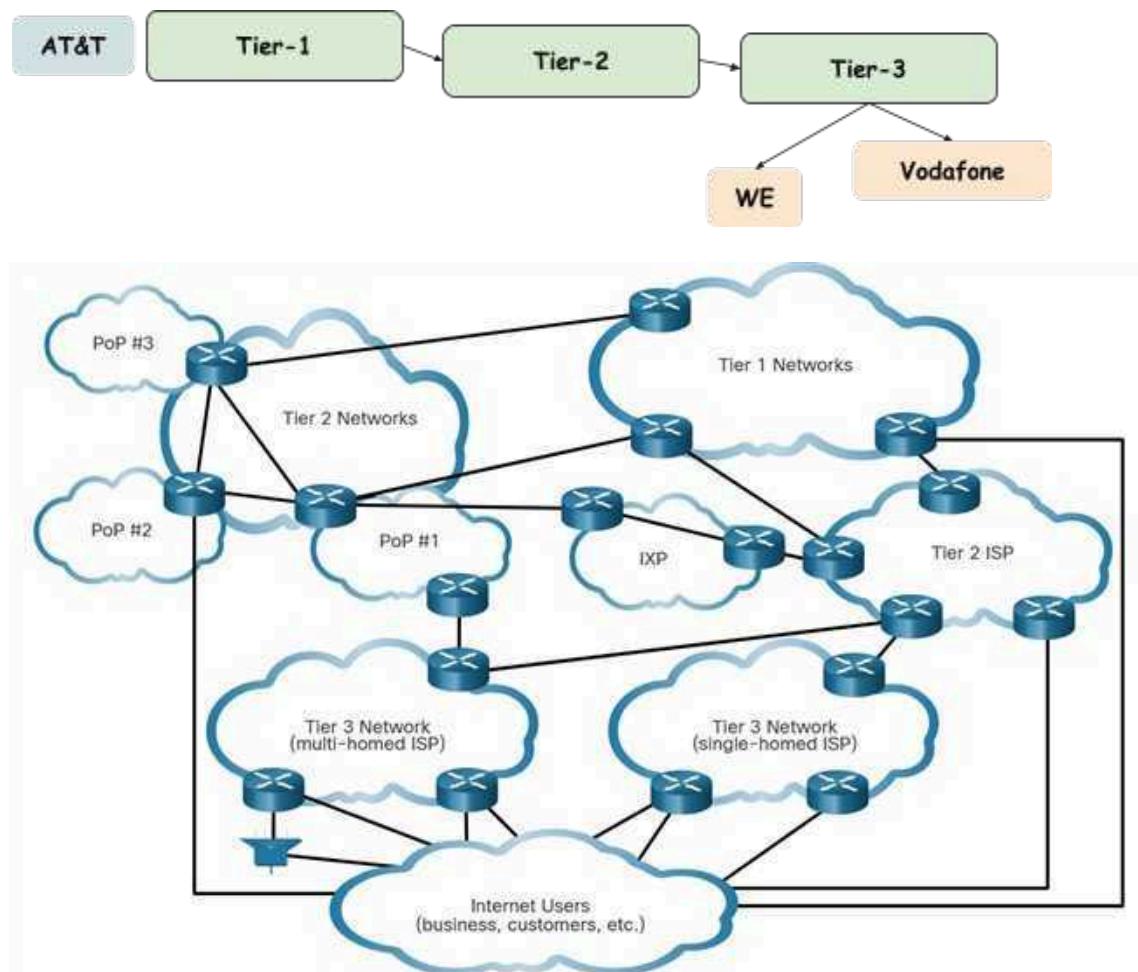
معظم شركات الـ ISP الكبيرة بتعتمد على بروتوكول Integrated IS-IS لانه سريع وبالتالي بيحافظ على استقرار الشبكة وكفاءتها.

- في مصر مش مشهور اسمه Noor شغال بـ Service Provider.
- Integrated IS-IS بتاعتها شغالة الـ Core منطقه الـ Vodafone شركة.

شركات الـ ISP على مستوى العالم مصنفة الى مستويات تعرف بالـ "Tiers"

- شركات الـ ISPs من Tire-1 زي AT&T و Verizon و NTT و دول من أكبر شركات الـ Telcom في العالم، يمتلكو البنية التحتية الأساسية للإنترنت على مستوى العالم (من كابلات بحرية، وأجهزة توجيه ضخمة ...). مش بتقدم خدمات الانترنت للعلماء والأفراد مباشرة .. ولكن بتتعامل مع الشركات الأصغر منها والمصنفة من Tier-2 .. والاتنين دول بيربطو أجزاء كبيرة من الانترنت بعضها من خلال مراكز Internet Exchange Point - IXP

- شركات Tier-2 بتربط الشركات الأصغر منها من Tire-3 من خلال Point of Presence - PoP هما اللي بيربطو شركات الـ ISP زي WE و Vodafone وغيرهم واللي بيقدموا خدماتهم للـ Business Home Users الصغيرة.



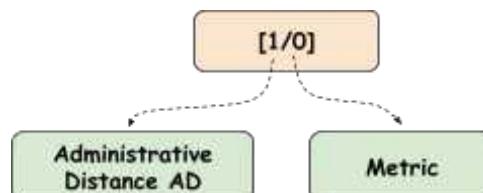
14.2.2. Distance Vector Routing Protocols (AD & Metric)

هنالاحظ وجود رقم زي [1/0] بجانب اي Route موجود في الـ Routing Table سواء Static او

- .Dynamic

```
R1#ip route 12.0.0.0 255.255.255.0 11.0.0.2
R1(config)#do show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
      + - replicated route, % - next hop override
Gateway of last resort is not set
      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.0.0.0/24 is directly connected, FastEthernet0/0
L        10.0.0.1/32 is directly connected, FastEthernet0/0
      11.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        11.0.0.0/24 is directly connected, FastEthernet0/1
L        11.0.0.1/32 is directly connected, FastEthernet0/1
      12.0.0.0/24 is subnetted, 1 subnets
S        12.0.0.0 [1/0] via 11.0.0.2
R1(config)#

```



Routing Protocol	Administrative Distance
Direct Connected	0
Static	1
EIGRP	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120

- أول رقم "1" اسمه الـ *Administrative Distance* وهو مقياس

لمصداقية الـ *Route* بالنسبة للراوتر (Measurement of Believability).

وكل ما كان اقل >> مصدقتيه تتبقى اعلى.

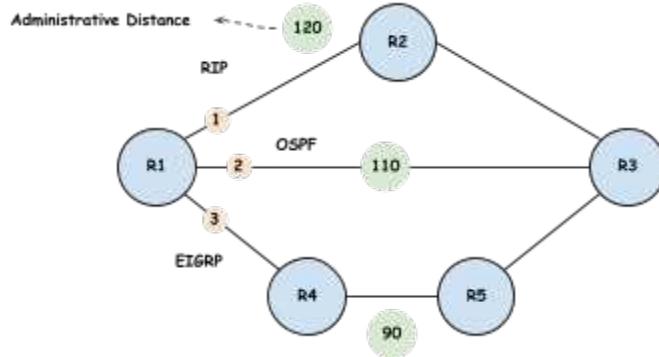
- كل *Protocol* له مقياس للمصداقية، بعضهم موضوع في

الجدول.

- قيم المصداقية دي يختلف من *Vendor* الى آخر، زي Cisco و Juniper •

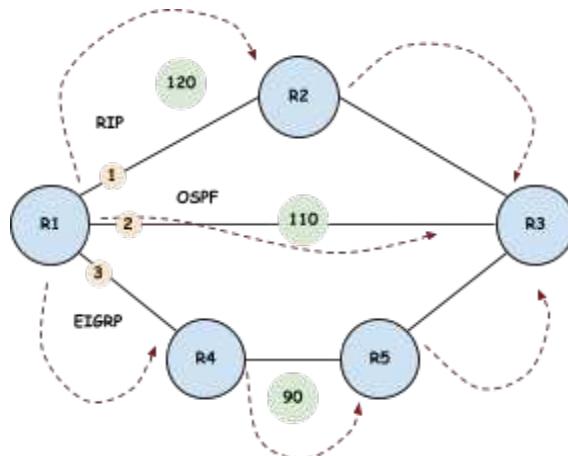
.Huawei

في الـ Topology التالي .. لو R1 عايز يصل لـ R3، وكل مسار شغال بـ RP مختلف.



- في الحالة دي R1 هيفضل المسار الثالث لأنه أفضل مصداقية.
- لو افترضنا إن كل المسارات شغالة بنفس الـ RP وهو RIP .. يعني كل المسارات ليها نفس الـ Administrative Distance او المصداقية اللي هو 120.
- هنا هيحصل "Tie" يعني عقدة .. واللي هيفرق بين الثلاث مسارات قيمة اسمها "Metric" يعني معيارية.
- كل RP له معيارية يعني طريقة تفكير .. مثلا الـ RIP بيفكر بطريقة الـ Number of Hops .Destination

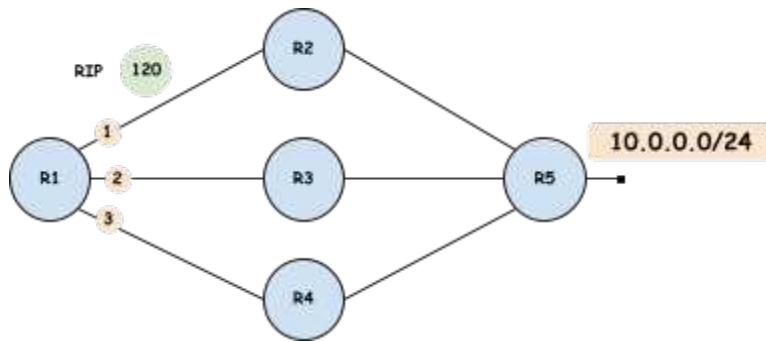
 - مسار 1 فيه اتنين Hubs يعني قفزيتين عشان يصل للـ Destination
 - مسار 2 فيه Hub واحد.
 - مسار 3 فيه ثلاثة Hubs.



- كدا الرووتر هيفضل مسار 2 لأن فيه عدد أقل من الـ Hubs .. والطريقة دي غبية بسبب ان المسار دا ممكن يكون هو الأقل سرعة.
- في RP بي Shawf المسار الأسرع، زي الـ OSPF .. او مثلًا EIGRP بي Shawf المسار الأسرع وبيتأكد انه على طول المسار مش هيحصل Delay.

Another Scenario

عندنا Topology بالشكل التالي .. فيها 3 مسارات، كلهم شغالين RIP، يعني لهم نفس الـ AD، وكل المسارات فيها اتنين Hub فقط .. وبالتالي عشان R1 يصل لشبكة 10.0.0.0



- اول حاجة هي Shawf الـ AD <> هيلاقيم متساوين
- وبعدين هي Shawf الـ Metric <> هيلاقي كلهم اتنين Hub
- هيلجاً لطريقة Equal Cost Load Balance <> يعني هيوزع الحمل على الثلاث مسارات .. فمثلا لو عندو 100 Packet، هيبيت شوية على المسار الأول وبعدين شوية على المسار الثاني وبعدين على المسار الثالث وهكذا.
- افتراضيا لو في اكتر من اربع مسارات <> هيختار أربعة فقط هيوزع الحمل عليهم.

14.3. Floating Static Route

الـ Floating Static Route هو نوع من أنواع الـ Static Routes، لكن بنغير فيه حاجة مهمة وهي الـ (AD). الفكرة كلها إننا بنسخدمه كـ Backup Route أو مسار احتياطي، يعني ما يشتغلش غير لما الطريق الأساسي يقع.

سبب استخدام الـ Floating Static Route

- الـ Static Route بيأخذ AD قيمته 1 (يعني موثوق فيه جداً).
- الـ OSPF زي Dynamic Routing Protocols بتاخذ AD أكبر (110 مثلاً).

وبالتالي الـ Router بيفضل الـ Static Routes عن اي اتعمله بشكل Route. الـ Dynamic Routes بنسخدمه في سيناريوهات لما نكون عايزين نقطط الـ static route بـ AD أعلى من الـ Route اللي احنا محتاجينه، عشان الرووتر يختار المسار دا ويفضله عن الـ Static Route.

فمثلا لو فيه طريقين، واحد عن طريق OSPF، والباقي Static Route. الرووتر هيختار الـ Static Route لأن الـ Static Route بتاعه = 1، وده مش اللي إحنا عايزينه. «إحنا عايزين نخلي OSPF هو الأساسي، والـ OSPF Route يشتغل كـ Backup Link يعني يتم استخدامه فقط لما يحصل مشكلة في الـ OSPF Route».

طريقة الـ Configuration

بنكتب نفس الامر المستخدم لتعريف Static Route مع اضافة الـ AD الجديد في الآخر، مع مراعاة انه يكون أعلى من المسار اللي احنا عايزين الرووتر يختاره.

```
R1(config)# ip route 192.168.2.0 255.255.255.0 192.168.3.1 130
```

```
R1# show ip route 192.168.2.0
Routing entry for 192.168.2.0/24
  Known via "ospf 1", distance 110, metric 20, type intra area
  Last update from 10.0.0.2 on GigabitEthernet0/0, 00:00:12 ago
  Routing Descriptor Blocks:
    * 10.0.0.2, from 10.0.0.2, 00:00:12 ago, via GigabitEthernet0/0
      Route metric is 20, traffic share count is 1
```

الأمر يعرض شوية معلومات عن الـ Route دا، ومنها ان الـ Router بيستخدم الـ Route اللي اتعلمته عن طريق الـ OSPF

لو الـ OSPF فيه مشكلة، هيستخدم الـ OSPF Route

```
R1# show ip route 192.168.2.0
Routing entry for 192.168.2.0/24
  Known via "static", distance 130, metric 0
  Routing Descriptor Blocks:
    * 192.168.3.1, via Serial0/0/1
      Route metric is 0, traffic share count is 1
```

حتى الـ Routing Protocols المختلفة ممكن تغير الـ AD الخاص بيها

في سؤال بييجي في امتحانات الشهادة .. بيسال على نوع الـ Route

```
R1# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is 10.0.0.1 to network 0.0.0.0

S*    0.0.0.0/0 [1/0] via 10.0.0.1
S      192.168.10.0/24 [1/0] via 10.0.0.2
S      192.168.10.0/24 [150/0] via 10.0.0.4
S      192.168.20.5/32 [1/0] via 10.0.0.3
```

مثلاً بيسال على نوع الـ Route دا نوع الـ Route

• 192.168.10.0/24 لان لو الراوتر عاييز يروح لاي IP في الشبكة

• هيسال 10.0.0.2

• تالت مسار عبارة عن Static Route لأنه Floating Static Route عادي (وعرفنا دا من خلال الرمز

• 5) لكن له $AD = 150$ اللي هو المفترض يكون 1

• رابع مسار عبارة عن Host Route لأنه بيوجه الـ IP واحد فقط، ودا واضح من خلال الـ

• 32 اللي هو Prefix

• اما اول مسار عبارة عن Default Route بيستخدم لو مفيش Route لشبكة او IP معين

15. RIP

بروتوكول RIP اختصار ل Distance Vector Routing Protocol وهو Routing Information Protocol .UDP على بورت 520 .RIP ممكن نتعامل معاه .. بيستغل في ال Application Layer

15.1. RIP Stages

عشان نفهم أي Routing Protocol بطريقة صحيحة .. لازم نفهم الـ 3 مراحل اللي بيمر بيهم:

- At Startup

يعني اول ما تشن بروتوكول RIP على الـ Router .. هيتصرف ازاي عشان يبني الـ Routing Table .Table

- At Convergence (aka stability)

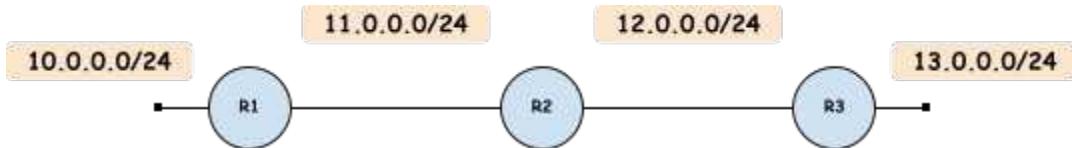
بعد ما الـ Routing Table اتبني خلاص وبقت الشبكة Stable .. ايه الـ Actions اللي .Routing Protocol بياخدتها الـ

- At Change

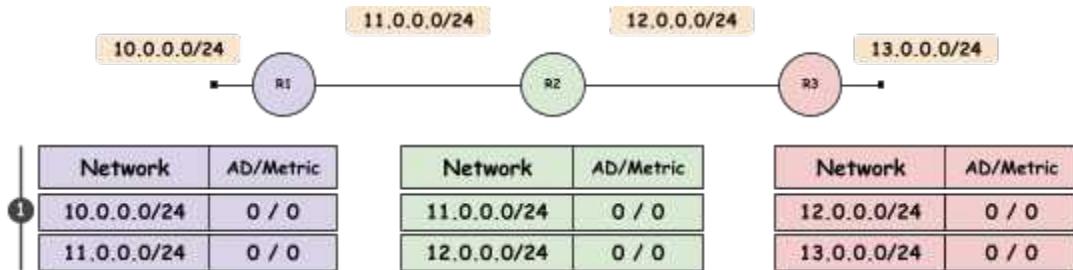
لو حصل تغيير (مثلا شبكة وقعت او اتضافت، راوتر وقع او اتضافت)، المفروض التغيير دا يتنقل لباقي الـ Routers .. مثلا لو شبكة اتضافت على Router .. المفروض تنضاف على الـ Routing Table بتابع الرواوترات الثانية.

15.1.1. At Startup

Let we have the following topology

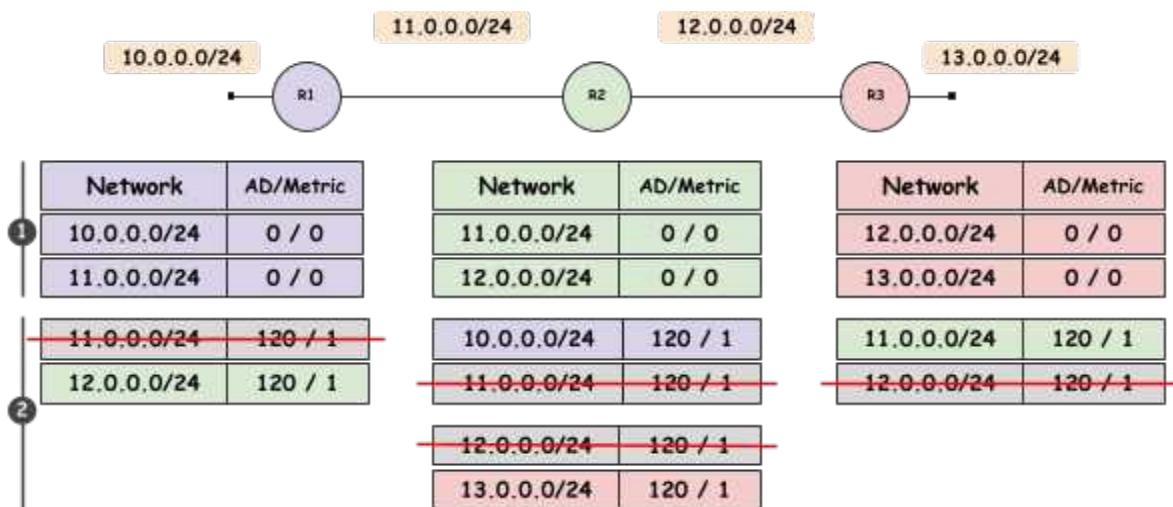
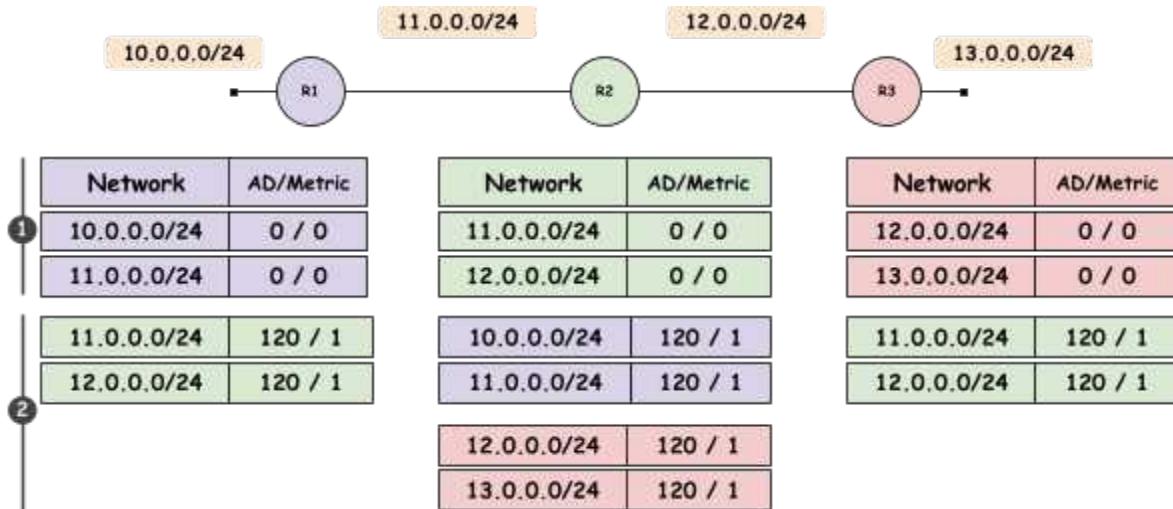


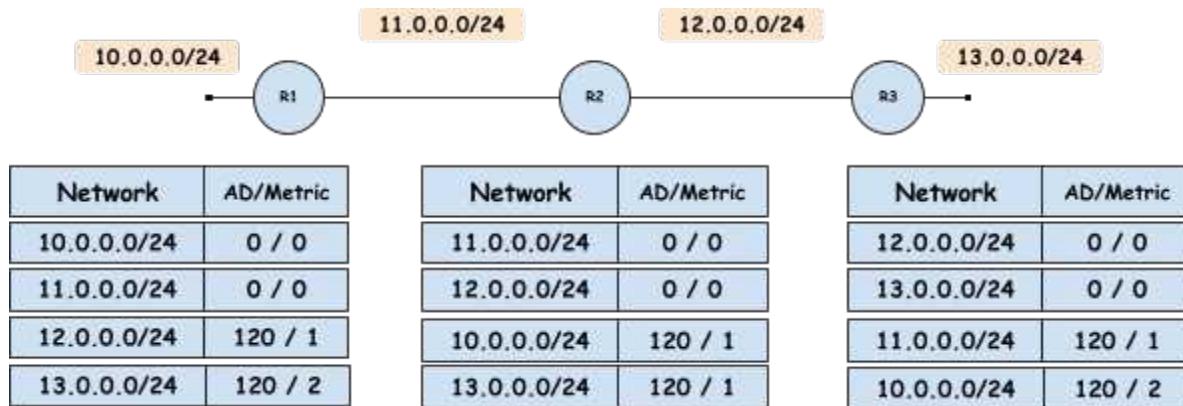
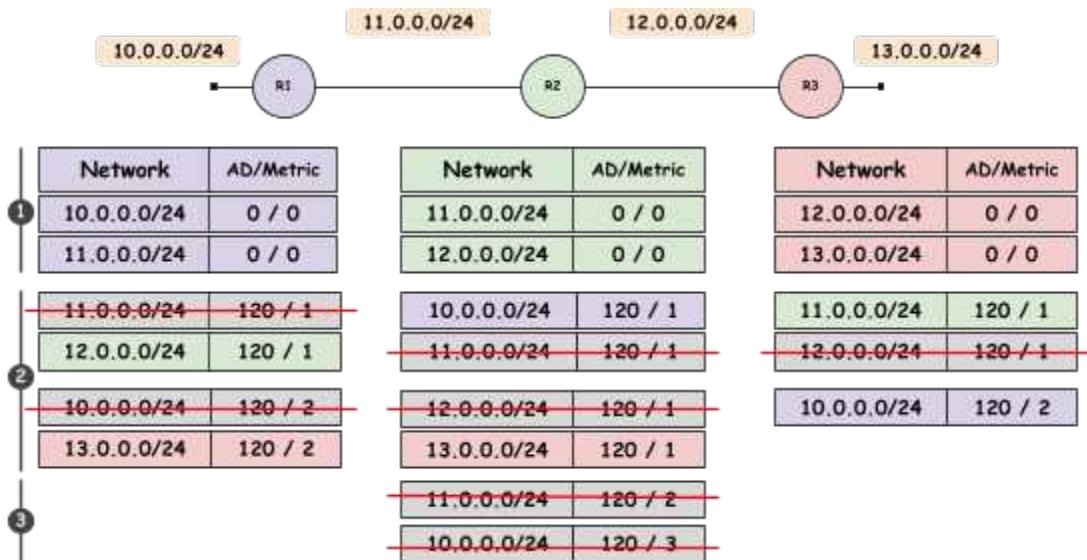
- بعد عمل IP Configuration للـ Router على الـ Port No Shutdown .. يبديا في اضافة الشبكات الـ Direct Connected في جدول التوجيه.
- بالنسبة للشبكات المتصلة مباشرة، يكون الـ Metric في جدول التوجيه صفر (0) لأن مفيش أي hop للوصول للشبكة. والـ AD يكون صفر للـ Direct Connected



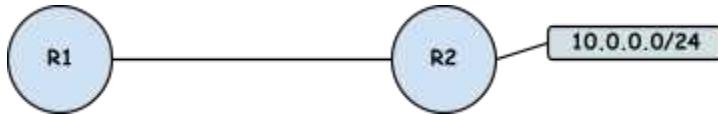
- **تحديثات RIP:** بروتوكول RIP بيزيغ الـ Routing Table مع جيرانه بالإشاعة "By Rumor" عشان يحدث جداول التوجيه في الراوترات الثانية.
- **قيمة الـ Metric:** لما راوتر يستقبل تحديث من راوتر جار، بيضيف معلومات الشبكة دي في جدول التوجيه بتاعه ويزود الـ metric بـ 1، عشان يعبر عن المسافة بقفزة واحدة (One Hop).
- **اختيار أفضل مسار:** لو الراوتر استقبل معلومات عن شبكة موجودة عنده، بيختار الشبكة اللي لها أقل ويحتفظ بيها في جدول التوجيه.
- لو المسارين لهم نفس الـ AD، الراوتر هيسخدم الـ Metric، عشان يقرر، يعني هيختار الـ Route اللي له أقل Metric.

- الراوتر يبحث جدول التوجيه باتجاه أفضل مسار (أقل AD ولو متساوين يبقى أقل Metric).





15.1.2. At Change



- R2 هيبيعت شبكة 10.0.0.0/24 لـ R1 بـ Metric=0، و R1 هيستقبلها ويزيود 1 وبالتالي هيشفوفها بـ "Metric = 1" .. وبعدين R1 هيبيعت نفس الشبكة تاني لـ R2 فهتوصله بـ 3 وهيعملها لانه شايفها Direct Connected اصلـ Drop.
- لما صمم بروتوكول الـ RIP خلى أكبر Metric للشبكة يساوي "15" اما 16 معناها ان الشبكة دي Unreachable وبنسميتها **Poisoned Route** يعني "طريق مسموم".
- وبالتالي لو شبكة الـ 10 وقعت، R2 هيبيتها بـ Metric=16 عشان تبقى Inaccessible بالنسبة لـ R1 .. بس لو افترضنا إن قبل ما الـ R1 بـ << كان R1 بيزيغ الـ Routing Table >> بتاعة R2 توصل لـ R1 بـ Packet عادي لأنها متسجلة عنده بـ Metric=16 بتاعو عادي فيبعتها بـ 2 .. كدا R2 هيقبل الـ Packet عادي لأنها متسجلة عندو بـ Metric=16 .. وهيفضل كل راوتر يبيعتها للثاني لحد ما توصل لـ Inaccessible وتبقى Metric=16 عند الأثنين، المشكلة دي اسمها **Continuing To Infinity**، وحل المشكلة دي هو الـ TTL .. ولكن ما زال عندي مشكلة Loop.
- في اسمها Split Horizon Feature حلت المشكلة دي، بتنمنع الراوتر إنه يبيعت معلومة عن شبكة معينة على نفس الـ Interface اللي استقبل منه المعلومة. وبالتالي، ده بيقلل احتمالية الـ Loops.
- في كمان اسمها Triggered Update With Poison Reverse Feature لما الشبكة تقع، راوتر R2 مثلًا مش هيستنى مدة 30 ثانية عشان يبيعت Update (زي ما بيحصل في الـ Poison Updates)، بل هيبيعت Triggered Update فوري لـ R1. وكذلك R1 هيبيعت لـ R2 ولباقي الـ Routers عشان يعرفوا ان الشبكة وقعت.

15.1.3. At Convergence

- **RIPv1** → will broadcast full routing table every 30 sec using the broadcast address 255.255.255.255
- **RIP v2** → will multicast full routing table every 30 sec using the multicast address 224.0.0.9
- **IGRP** → will broadcast full routing table every 90 sec

مشكلة RIPv1 إنها "رغاي" بيعت الـ Broadcast كل 30 ثانية Routing Table يعني بتوصل للكل سواء أو **Routers**، وكمان شغال في الـ App Layer يعني لما الـ Packet توصل لـ Dest هيعملها **Clients** لحد ما يوصل لـ App Layer ودا بيستهلك الـ CPU اكتر.

ومشكلته كمان إن الـ Packet مش بتشيل غير Route 25 يعني لو في Route 150 هيتتم تقسيمهم على 6 يعني كل 30 ثانية هتبعد 6 Packet فدا بيعمل Load كبير على الـ Processor والشبكة.

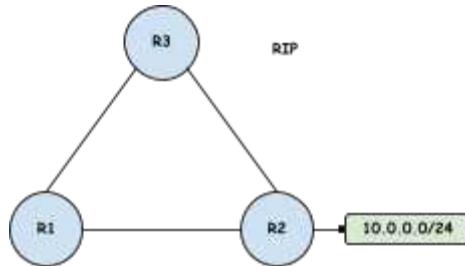
RIPv2 حل مشكلة الـ Broadcast وبقى بيعت الـ Multicast بطريقة Routing Table يعني كل الـ Routers يبي Listen على الـ Multicast Address: 224.0.0.9 اللي بتـ.

بروتوكول **IGRP** هو بروتوكول توجيه تم تطويره بواسطة Cisco في الثمانينيات كبديل لبروتوكول RIP، بهدف تحسين أداء التوجيه في الشبكات الكبيرة ومعالجة القيود الموجودة في RIP. يعني مثلاً أقصى عدد لـ Hops ييوصل لـ 100 بشكل افتراضي وممكن يصل لـ 255، وكمان بيعت تحديثاته كل 90 ثانية، بدل كل 30 ثانية في RIP، وده بيقلل من الـ overhead على الشبكة. والـ Metric بتاعه بيعتمد على سرعة المسار وكل المميزات دي خلته بطيء ويستهلك Resources بشكل كبير زائد الـ Delay Factor .. وكمان ملكية خاصة بـ Cisco.

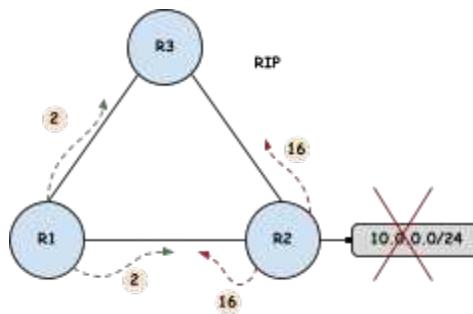
15.2. Split Horizon And Holddown Timer

في حالة الـ **Redundant Topologies** .. فمثلا لو عندي السيناريو التالي:

- شبكة فيها 3 راوترات (R1, R2, R3)، وشبكة متوصلة براوتر R2، والثلاثة Routers 10.0.0.0/24.
- واصلين بعض.



- لو شبكة 10.0.0.0/24 وقعت <> R2 هيبعثت المعلومة دي بـ (Metric = 16) للراوترين R1 و R3.
- المشكلة: لو R1 مثلا بعثت المعلومة دي -قبل ما يوصله ان الشبكة وقعت- لـ R2 و R3 (بـ (Metric=2
- زي ما قلنا ان الراوترات بتقبل الـ Metric الأقل، فهياصدقوا ان الـ شبكة لسا ما وقعتش .. ودا هيسبب فالشبكة Loop.



حل المشكلة دي في ميكانيزمين يعتمد عليهم بروتوكول الـ RIP

- يمنع الراوتر إنه يبعث معلومة عن شبكة معينة على نفس الـ Interface.
- استقبل منه المعلومة. وبالتالي، ده بيقلل احتمالية الـ Loops.

لما الرووتر يوصله Route ويكشف إن الشبكة وقعت، بيستنى **Holddown Timer** •
فترة معينة (180 ثانية By Default) قبل ما يقبل أي تحديثات بتقول إن الشبكة رجعت. وده بيمنع
الرووتر من اتخاذ قرارات بناءً على معلومات قديمة أو غير صحيحة. بس لو نفس الرووتر اللي بعتلي ان
الشبكة وقعت « بعتلي تاني ان الشبكة رجعت قبل الا 180 ثانية » هقبلها عادي.

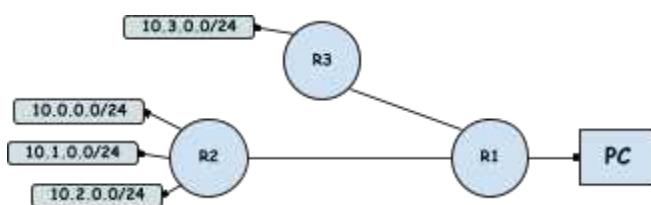
15.3. RIPv1 vs RIPv2

دعم الـ VLSM Variable Length Subnet Mask

RIPv1 •: مش بيدعم الـ VLSM، بمعنى إنه بيتعامل مع الشبكات بنظام الـ Classful Addressing اللي هو (Class A, B, C) فقط. يعني ميقدرش يستخدم أكثر من Major Subnet Mask في نفس الـ Network .. وبال التالي لو في اكتر من شبكة من نفس الـ Class متوصلين براوتر شغال بـ RIPv1 .. هيبيت الـ Major Network فقط، مش هيبيت كل شبكة بالـ Prefix بتاعها .. ولو راوتر تاني استلم المعلومة هيضيف لها الـ Default Prefix .

RIPv2 •: بيدعم الـ VLSM ويشتغل بنظام Classless Addressing، وده بيسمح باستخدام Subnet Masks مختلفة ومرونة أكبر في تقسيم الشبكات.

في المثال التالي، لو شغال بـ RIPv1 .. كل راوتر هيبيت الـ Major Network فقط (اللي هي 10.0.0.0 والراوتر اللي هيستلمها، هيسجلها بالـ Default Prefix للشبكة دي واللي هي 8 / .. وبال التالي لو الـ PC عمل Ping مثلًا على شبكة 10.2.0.0 <> R1 مش هيعرف يوجه الـ Packet دي لانه شايف الشبكة على اكتر من منفذ.



في حالة واحدة R2 هيبيت الـ Detailed Network لـ R1 .. وهي لو الشبكة اللي بين R1 و R2 من نفس الـ Major Network

اـ Routing Advertisements

- **RIPv1**: بيعت تحدیثات الـ Routing على الـ IP (عنوان General Broadcast IP) . وده بيستهلك الـ Bandwidth ويوصل التحدیثات لكل الأجهزة، حتى لو مش مهمـة.
- **RIPv2**: بيستخدم Multicast (عنوان 224.0.0.9) علشان بيعت التحدیثات للأجهزة المهمـة فقط، وده بيقلل الضغط على الشبـكة.

الأمان (Authentication)

- **RIPv1**: مفيش دعم للأمان أو المصادقة. أي جهاز يقدر بيعت تحدیثات RIPv1.
- **RIPv2**: بيـدعم الـ Authentication عن طريق Plain-Text أو MD5، وده بيضيف طبقة من الحماية لتأمين تحدیثات الـ Routing.

التوافقية مع الشبـكات القديمة

- **RIPv1**: بيشتغل فقط مع الشبـكات اللي بتستخدم العناوين التقليدية (Classful).
- **RIPv2**: متـافق مع RIPv1، لكن بيـضيف تحسـينات علشـان يتعامل مع الشـبـكات اللي بتـستخدم Classless Addressing.

الـ Control Plane هي الـ Updates اللي بتتنقل بين الـ Routers والـ Routing Table . اما الـ Data Plane هي البيانات اللي عايـزين نقلـها زي الـ Ping والـ Share .. يعني الـ Control Plane بـعتمد على الـ Data Plane .

15.4. RIP Futures

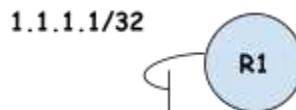
The Routing Information Protocol (RIP) is a distance-vector routing protocol used in computer networks. Here are some properties and characteristics of the RIP protocol:

- **Distance-Vector Algorithm:** RIP uses a distance-vector algorithm to determine the best path or route to a destination network. It uses Hop Count as its Metric to determine the best route. Each router increments the hop count as a packet passes through it. The maximum hop count value in RIP is 15, meaning that a route with a hop count greater than 15 is considered unreachable and limiting its scalability in larger networks.
- **Split Horizon:** RIP employs a split horizon mechanism to prevent routing loops. It does not advertise routes back to the same interface from which they were learned, ensuring that routing information is not sent back to the originating router.
- **Poison Reverse:** To further avoid routing loops, RIP uses poison reverse, where a router advertises a route with an infinite metric (hop count of 16) back to the router from which it learned the route. This informs the originating router that the route is unreachable.
- **Periodic Updates:** RIP periodically sends routing updates to neighboring routers. By default, RIP sends updates every 30 seconds, which can result in increased network traffic and convergence time.
- **Classful Routing:** RIPv1 is a classful routing protocol, which means it does not carry subnet mask information with its route advertisements. It assumes that all networks within a classful network have the same subnet mask.
- **Classless Routing:** RIPv2 supports classless routing, which means it can advertise network prefixes along with their subnet mask information.
- **Convergence Time:** Convergence time refers to the time it takes for the routing tables to stabilize after a change in the network topology. Convergence can be slower in RIP compared to other routing protocols due to its periodic update mechanism.

- **Limited Network Size:** Due to its hop count limitation and periodic updates, RIP is best suited for small to medium-sized networks. It may not scale well in larger networks or environments with frequent topology changes.
- **Authentication:** RIPv2 supports authentication mechanisms, such as MD5 authentication.
- **VLSM:** RIPv2 supports Variable-Length Subnet Masking (VLSM), also supports route summarization (CIDR), and the ability to carry multicast and unicast routing information.
- **Backward Compatibility:** RIPv2 routers can communicate with RIPv1 routers, but the RIPv1 routers will only receive classful updates without subnet mask information.
- **Route Tagging:** RIPv2 supports route tagging, which allows the administrator to assign additional information to specific routes. Route tags can be used for administrative purposes, such as indicating the origin or priority of a route.

15.5. RIP Configuration

لتوضيح ان في Loopback Interface على الرسم، بنستخدم الطريقة دي



دائماً بنستخدم رقم الراوتر لعمل الا IP بتاعه بيكون بنفس الرقم للتسهيل .. يعني مثلاً R1 ممكن اعمل عليه Loopback Interface رقمه 1 باستخدام أمر 1 loopback Interface وهاخد Prefix IP:1.1.1.1 وال IP 32 لانه واحد فقط.



اول حاجة المفروض نعمل لل Loopback Interface Configuration

```
R1(config)# int loopback 1
R1(config-if)# ip add 1.1.1.1 255.255.255.255
```

```
R2(config)# int loopback 2
R2(config-if)# ip add 2.2.2.2 255.255.255.255
```

وبعدين نعمل لل IPs Configuration

```
R1(config)# int f0/0
R1(config-if)# ip add 10.0.0.1 255.255.255.0
```

```
R2(config)# int f0/0
R2(config-if)# ip add 10.0.0.2 255.255.255.0
```

لعمل Configuration لبروتوكول RIP عشان نخلي R1 يشوف شبكة 2.2.2.2/32 على R2 .. و R2 يشوف

شبكة 1.1.1.1/32 على R1

عرض الـ Router الموجودة على الـ Routing Protocols

```
R1# show ip protocols
```

لتشغيل بروتوكول RIP بنتطبق الأمر rip في الـ Configuration Mode في الـ Routerrip وبعدين لتطبيق البروتوكول على Interfaces معينة .. بنحدد الشبكة الموجودة على الـ Interfaces دي وبعدين الـ Router يعمل علىInterfaces لـ Network اللي عليها IP من نفس الـ Network اللي حدتها Match

```
R1(config)# router rip
R1(config-router)# network 1.1.1.1
R1(config-router)# network 10.0.0.0
```

```
R2(config)# router rip
R2(config-router)# network 2.2.2.2
R2(config-router)# network 10.0.0.0
```

لو كتبت امر

```
R1# show ip protocol
```

هيظهر ان الـ Router بيرسل الـ Routing Table بـ V1 و يستقبل بـ V1,2 افتراضيا .. لكن لو حددت معين، هيرسل ويستقبل بالـ Version دا فقط.

لتحديد Version معين

```
R1(config-router)# version 2
```

ملحوظة: اصدار 1 و 2 مش Compatible مع بعض .. يعني لو في راوترین شغالين كل واحد شغال باإصدار مختلف >> مش هيفهمو الـ Routing Table اللي بيتعته البعض.

- المفروض ان الـ Routing Table يتبع كل 30 ثانية
 - في حالة ان كل Router شغال باصدار مختلف <> الـ Timer هيكمل لـ 180 ثانية وبعدين يظهر ان الشبكة بقت Passably down .. ولكن هفضل موجودة في الـ Routing Table
 - وبعدين الـ Timer هيكمل لـ 240 ثانية والشبكة هتمسح من الـ Routing Table
- لـ Troubleshooting ممكن نستخدم امر Debug اللي بيعرض الـ Output في الـ Realtime بعكس امر Show اللي بيعرض الـ Output في الوقت الحالي فقط

```
R1# debug ip rip
```

لكن في بعض الـ Options في امر Debug بتعمل Load عالي على الـ CPU .. وممكن يهنج الـ Router .. وفي الحالة دي لازم اعمل Reload.

لالغاء امر Debug

```
R1# no debug ip rip
```

لالغاء كل اوامر الـ Debugging اللي شغالة

```
R1# undebug all
```

لتحذف الـ Routing Table

```
R1# clear ip rout *
```

الامر دا خطير لانه بيعمل Clear لـ Routing Table ويبدأ بيئيه من جديد .. ولازم تتجنب استخدامه في البيئة (Production).

15.5.1. RIP Summarization

الـ Default في اصدار RIPv1 و 2 انه ي يعمل الـ Auto Summarization يعني يبعث الـ Auto Summarization .. لازم نوقف خاصية More Detailed Network .. عشان تخلي الـ RIPv2 يبعث الـ عن طريق الامر:

```
R1(config-router)# no auto-summary
```

المشكلة دلوقتي لو عندي شبكات كتير جدا .. 200 الف Route مثلا .. ودا هيكيبر الـ Routing Table بشكل كبير، فخاصية الـ Summarization مفيدة في الحالة دي

بس مشكلتها ان لو عندي شبكات كتير من نفس الـ Subnet مش هتظهر في الـ Routing Table .. فانا بين ميزة وعيوب .. انها بتختصر الـ Routing Table، وانها مش بتظهر كل الشبكات

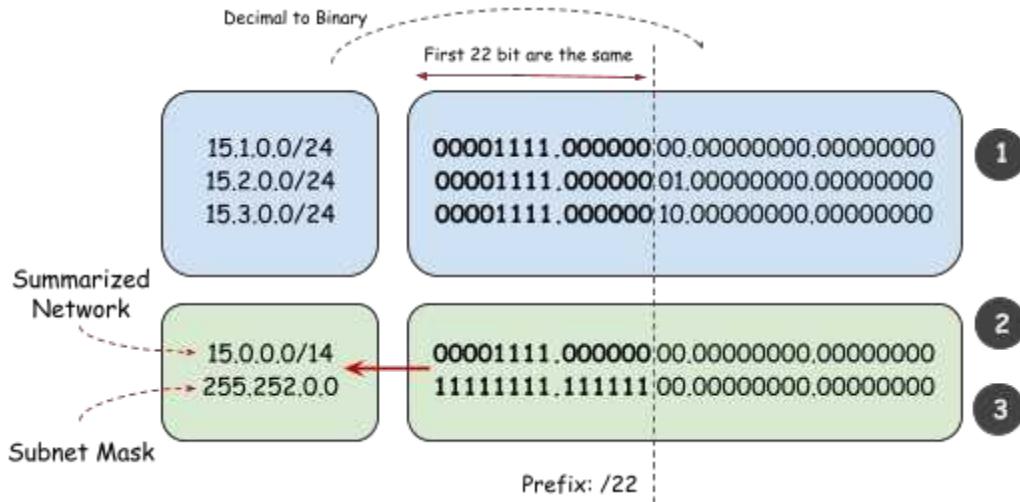
بس في خاصية افضل وهي انني ممكن اعمل Manual Summarization للشبكات الموجودة على Interfaces معينة

مثلا لو عندي الشبكات دي $15.3.0.0/24$ - $15.2.0.0/24$ - $15.1.0.0/24$ ممكن اختصرهم في الشبكة $15.0.0.0/14$ عن طريق الامر

```
R1(config-router)# ip summary-address rip 15.0.0.0 255.252.0.0
```

للوصول للـ Summarized Network .. اول خطوة بنحول كل شبكة الـ Binary

1. حول كل الشبكات الى Binary
2. لحساب الـ Summarized Network <> خلي الجزء المتشابه بدون تغيير، وحول كل الجزء المختلف من اليمين الـ Zeros، وبعددين رجع الرقم الى Decimal
3. لحساب الـ Subnet Mask <> حول الجزء المتشابه الى Ones والمختلف الى Zeros ورجع الرقم الى Decimal



15.5.2. RIP Authentication



باستخدام الـ Authentication .. الراوترات مش بتقبل اي Updates الا لو جاية من Routers بتستخدم نفس الـ Password اللي حددته.

- ممكن اعمل اكتر من مفتاح او Key في سلسلة مفاتيح (Keychain) بيكون لها اسم محدد .. وكل مفتاح فيها له رقم وتاريخ ابتداء وتاريخ انتهاء، وممكن اضيف اسم اختياري.
- ممكن اعمل اكتر من Keychain مع تخصيص كل واحدة لـ Interfaces معين.
- الـ Interface بيتعملها Keychain في الـ Config Mode في الـ Configuration وبعدين بتنطبق على الـ Configuration.
- اسم الـ Keychain بيكون Keys بس الـ Keys نفسها لازم تكون متطابقة على الـ Routers اللي هتشتغل مع بعض.

لإنشاء Keychain

```
R1(config)# key chain TSHOOT
R1(config-keychain)# key 1
R1(config-keychain-key)# key-string CCNA
R1(config-keychain-key)# accept-lifetime 00:00:00 1 jan 2022 00:00:00 30 june 2022
R1(config-keychain-key)# send-lifetime 00:00:00 1 jan 2022 00:00:00 30 june 2022
R1(config-keychain-key)# exit
```

في مجموعة الأوامر دي .. أنشأت Keychain اسمها TSHOOT وعملت فيها Key رقم 1 وبعدين اضفت اسم اختياري لـ Key 1 وهو CCNA ثم حددت الـ Accept Lifetime وهو تاريخ قبول الـ Key وحددت الـ Send Lifetime وهو تاريخ بدأ وانتهاء الـ Key.

لتطبيق الـ Interface على Keychain معين

```
R1(config)# int f0/0
R1(config-if)# ip rip authentication key-chain TSHOOT
R1(config-if)# ip rip authentication mode <text or md5>
```

الـ Mode يحدد طريقة تشفير الـ Key سواء Text اللي بيعت الباسورد بشكل Plaintext، وده غير آمن لأنه ممكن يتم اعتراضه بسهولة. او MD5 اللي بيستخدم تشفير MD5 Hash، وده أفضل وأكثر أماناً لأنه مش بيعت الـ Key مباشرة، بل بيعت الـ Hash الخاص بي.

تنوية:

- لو اخترت Md5، لازم رقم الـ Key يكون متطابق على الروائزات اللي هتطبق Authentication عليها لأن رقم المفتاح بيستخدم في عملية التشفير.
- لو في عدم تطابق في الـ Authentication وعملت Debug المشكلة هتظهر "Authentication Invalid" .. أما على الـ OSPF بيظهر نوع المشكلة بالضبط في الـ "Authentication".

15.5.3. RIP Equal Cost Load Balance

في بروتوكول RIP .. لو الراوتر استلم Route موجود عنده في ال Routing Table بنفس ال AD Metric والـ Load Balance .

مثلا لو عندي نفس الشبكة باكتر من مسار، هتظهر بالشكل دا

```
R1# show ip route rip | begin 4.4.4.4
R        4.4.4.4 [120/2] via 10.0.0.3, 00:00:19, GigabitEthernet0/2
                  [120/2] via 11.0.0.2, 00:00:08, GigabitEthernet0/1
```

بروتوكول RIP ي يعمل Load Balance على أربع مسارات فقط By Default والاعداد دا بيظهر في ال Config

```
R1# show running-config all | begin router rip
router rip
version 2
validate-update-source
timers basic 30 180 180 240
network 11.0.0.0 mask 255.255.255.0
network 10.0.0.0 mask 255.255.255.0
maximum-paths 4
input-queue 150
distance 120
```

ممكن نغير الاعداد دا ونخلية يختار مسار واحد فقط (وبالتالي مش هي عمل Load Balance)

```
R1(config)# router rip
R1(config-router)# maximum-paths ?
      <1-32> Number of paths
R1(config-router)# maximum-paths 1
```

وبعد تطبيق الامر دا «» هيختار المسار صاحب أعلى Next Hop IP Address

```
R1# show ip route rip | begin 4.4.4.4
R        4.4.4.4 [120/2] via 192.168.13.3, 00:00:07, GigabitEthernet0/2
```

ولو المسار دا وقع، هيرجع يضيف المسار الثاني

لتغيير الـ AD الخاص بـ Static Route معين

```
#R(config)# ip route 2.2.2.2 255.255.255.255 s1/0 121
```

Where 121 is the new administrative distance

لتغيير الـ AD الخاص ببروتوكول RIP

```
R1(config)# router rip  
#R(config-router)# distance 95
```

مصطلح الـ Scalability يعني ازاي اقدر اعمل Extend للشبكة بطريقة سهلة

Static Routing اعلى من الـ Scalability له Dynamic Routing - -

15.5.4. RIP Troubleshooting:

```
# show ip protocols (version, timers, networks, authentication)
```

```
# show ip rip database (check if the network is in the RIP database)
```

```
# show ip route rip
```

<https://networklessons.com/rip/troubleshooting-rip>

16. OSPF

هو Dynamic Routing Protocol يصنف تحت الـ Interior Gateway Protocols يعني يستخدم داخل الـ

Autonomous Systems المش بين Autonomous Systems المختلفة، وهو من فئة الـ Link State.

- الـ OSPF هو Open Standard Protocol له AD: 110.

يستخدم Shortest Path First - SPF أو Dijkstra Algorithm لتحديد المسار الأفضل اسمها.

للوصول للشبكة.

هو Reliable protocol يستخدم طرق خاصة لحفظ البيانات بعيدا عن الـ TCP لأنه مش

يتغلب في TCP ولا UDP.

16.1. OSPF Stages

16.1.1. At Startup

المرحلة دي مقسمة لاربع مراحل وهم Database Exchange و Neighbor Discovery ثم بناء Tree.

للمسارات الموجودة لكل شبكة وبعد بناء الـ Routing Table باختيار افضل مسار لكل شبكة.

16.1.1.1. Neighbor Discovery

مرحلة Neighbor Discovery هي أول خطوة في بناء علاقه Neighbor Adjacency او

Relationship بين الروابط اللي بتشتغل بالـ OSPF عشان يتداولوا معلومات التوجيه.



أول حاجة الـ Router بيكون في الـ Down State .. يعني لسا مشغل الـ OSPF وببيدا يعمل.

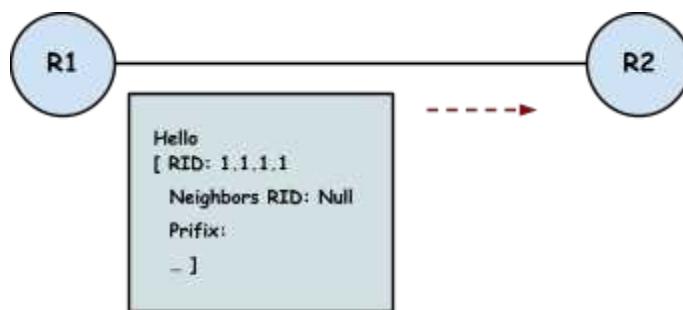
. Router ID - RID Generate وهو رقم مميز لكل Router في شبكة الـ OSPF.

الـ Router بيبدأ بإرسال رسالة اسمها Hello Packet على الـ Multicast IP: 224.0.0.5 (على الـ

Interfaces التي فعلت عليها الـ OSPF).

- الـ Hello Packet دي يحتوي على مجموعة من المعلومات:
 - Router ID: رقم مميز لكل راوتر في شبكة الـ OSPF
 - Subnet Mask: وهو الـ Network Mask
 - Hello Interval: الـ Router كل 10 ثواني يبعث Hello Packets افتراضياً ودي الـ Hello Interval.
 - Dead Interval: لو الـ Router مستقبلش Hello، بيستنى فترة الـ Dead Interval، وبعد كدا بيعتبر الجار (neighbor) غير متاح.
 - Area ID: رقم المنطقة اللي الـ Router فيها، لأن شبكة الـ OSPF بتقسام الى مناطق.
 - Authentication Data: لو الشبكة بتستخدم مصادقة.
 - Priority: الـ RID الخاص بالـ DR
 - Designated Router << DR
 - Backup Designated Router << BDR
 - Neighbors RID الخاص بالـ Neighbors

نفترض ان R1 له R1 RID:2.2.2.2 .. و R2 له R2 RID:1.1.1.1 بـ Hello فيها المعلومات الخاصة بيـه



- مجرد ما R2 يستقبل Hello من R1 -> كدا R1 هينتقل إلى حالة اسمها INIT State .. و R1 لو استلم Hello هينتقل إلى حالة الـ INIT State بـ رضـو.

- لما الرووتر يستقبل البيانات اللي في الـ **Hello Packet**. لو البيانات اللي في الـ **Packet** متوافقة معاه، الرووتر بيعتبر الرووتر **Neighbor** (جار). طب ايه الاعدادات اللي لازم تكون متوافقة او ايه شروط الـ **Neighbor**

:Relationship

لازم يكونو **Direct Connected**

لازم يكون لهم نفس الـ **Network ID** والـ **Subnet Mask**

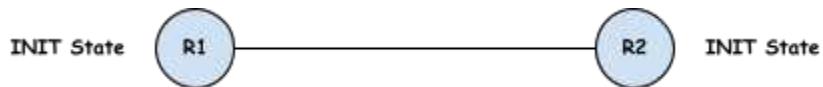
لازم يكون لهم نفس الـ **Hello Interval**

لازم يكون لهم نفس الـ **Dead Interval**

لازم يستخدمو نفس نوع الـ **Authentication** لو مطبق

لازم يكونو في نفس **Area ID**

Must agree in the same Flag



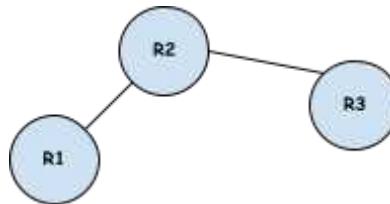
- بعد كدا R2 هيرد على R1 بـ **Hello** .. واللي هيكون فيها الـ **RID** الخاص بيه والـ **RID** الخاص بـ R1 اللي استلمه قبل كدا .. وكذا R1 هيبي في حالة الـ **Two-Way-Communication** لأنها استلم **Hello** بـ **RID** بتاعه.

و R1 هيرد على R2 بـ **Hello** برضو والاتنين هيبيقو في نفس الحالة



بنستفيد من الحالات دي في الـ **INIT State**، فمثلا لو الـ **Router** فضل في حالة الـ **Troubleshooting** نقدر نستنتاج ان في مشكلة في الـ **Channel** اللي بتسقبل عليها البيانات.

في السيناريو دا مثلا .. عشان الـ Routing Table ينتقل بين R1 و R2 .. لازم R1 و R3 يكونو Neighbors، و R2 و R3 يكونو Neighbors وبالتالي الـ Routing Table هينتقل بين R1 و R3 من خلال R2

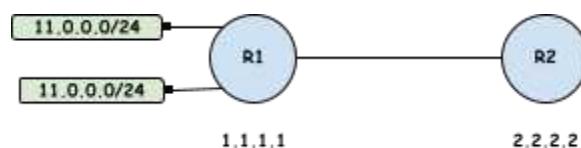


16.1.1.2. Database Exchange

بعد ما الرواوترات تتعدي مراحل Neighbors Two-Way State و Neighbor Discovery ويكونوا في مرحلة Fully Adjacent Neighbors عشان يكونوا Database Exchange . الهدف الرئيسي من المرحلة دي هو إن الرواوترات تتبادل قواعد البيانات (Link State Database - LSDB) عشان يتأكروا إن عندهم نفس نسخة معلومات التوجيه.

1. الخطوة الاولى: تحديد الـ Master والـ Slave

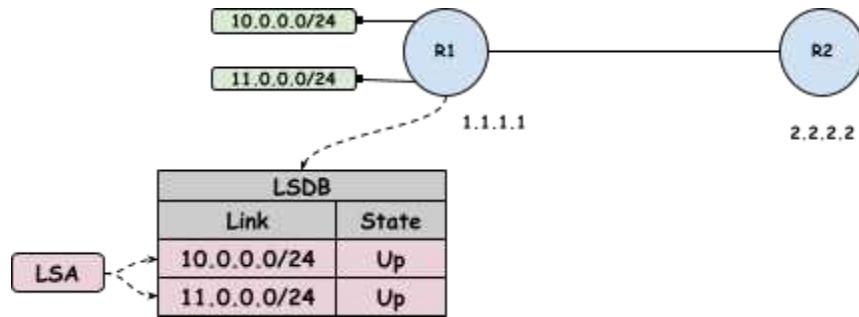
- الرواوترات بتحدد مين اللي هيبارد (Master) ومين اللي هيستجيب (Slave) عن طريق مقارنة الـ Router IDs اللي عرفوها عن بعض نتيجة تبادل الـ Hellos .
 - الرواوتر اللي عنده Router ID أكبر بيكون الـ Master
 - الـ Master هو اللي بيبدأ عملية تبادل الـ DBD Packets، والـ Slave بيرد عليه. العلاقة دي عشان ينظموا مين يبدأ ومين يرد أثناء تبادل المعلومات. وكل راووترين بيعملوا العلاقة دي مع بعض بشكل مستقل عن علاقتهم مع الرواوترات الثانية.



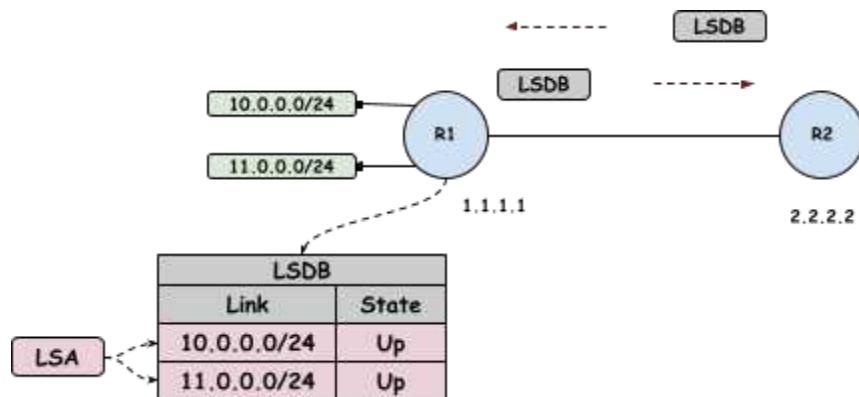
- المرحلة دي اسمها ExStart State

2. الخطوة الثانية: تبادل الـ DBD Packets

- كل راوتر يبقى عندو Link State Database - LSDB واللي بتكون من الشبكات اللي على الراوتر (فيها Advertisements - LSA) هي عبارة عن بيانات عن الشبكات اللي على الراوتر (فيها الشبكة وحالة الـ Interface بالإضافة إلى مجموعة من البيانات)



- كل Router يبعث DBD Packets لباقي الراوترات .. واللي بيكون فيها Summary عن المعلومات الموجودة في الـ LSDB، او ممكن نقول فيها الـ LSAs Headers
- الهدف من تبادل الـ DBD هو ان كل Router يعرف معلومات عن الشبكات اللي ناقصاه .. والحالة دي اسمها ExChange State

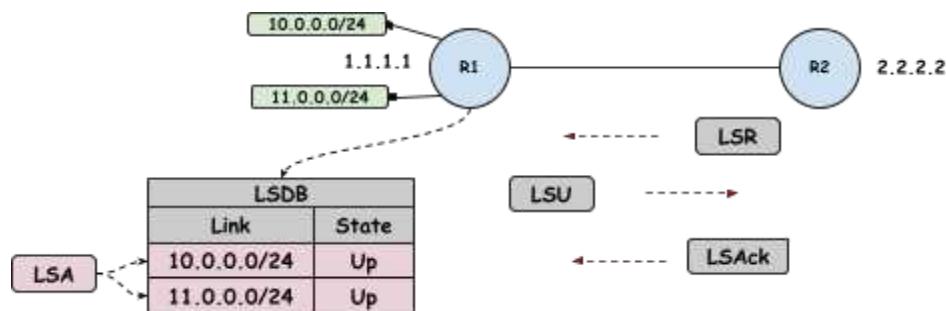


- بعد تبادل الـ DBD .. كل Router ي يكون عرف الشبكات اللي مش موجودة عنده في الـ LSDB

- وبيبدا يطلبها عن طريق ارسال Link State Request - LSR

- والراوتر الثاني بي رد عليه بـ Link State Update - LSU فيها المعلومة الكاملة عن الشبكة

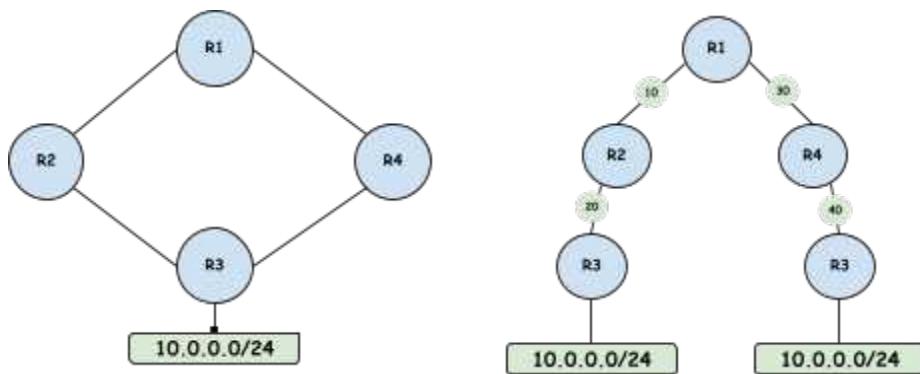
- وبعدين يرد عليه بـ Link State Ack للتاكيد



- والحالة دي اسمها Loading State

- بعد ما كل راوتر يعرف كل الشبكات اللي ناقصة ويطلبها، بينتقلو أخيراً إلى الـ LSDB وكل الراوترات بيكون عندها نفس الـ Neighbors.

- الراوتر دلوقتي بيبدا يستخدم الـ Dijecistra Algorithm (اسمها برضو SPF) عشان ببني الـ Routing Tree .. طريقة عملها ان كل Router بيبني Tree بكل المسارات الموصلة إلى شبكة معينة وبناء على معادلة الـ Cost والـ SPF الناتج عنها لكل مسار، بيخختار المسار صاحب أقل Cost.



16.1.2. At Convergence

- The router will send LSA refresh each 30 minutes

في حالة الـ **Convergence** كل Router بيعت LSA بشكل دوري كل 30 دقيقة، والخطوة دي مش مهمة اوكي، عشان كدا لو حصل تغيير « بيتبعت LSU » بشكل لحظي .. فمتش لازم كل شوية بيعت Trigger Update، فممكنا الـ Bandwidth .. بس الخطوة دي اتعملت عشان زمان كان الـ LSA ميوصلش، فكان بيعمل Refresh كل 30 دقيقة. بروتوكول الـ EIGRP لا يحتوي على الخطوة دي.

- The router will send a hello as a keepalive each hello interval

يعني الراوترات هيفضلو مستمرین في ارسال الـ Hello Packets

تنويه: الراوتر دائمًا بيعت LSU والي بتكون من اكتر من LSA .. وبالتالي لما بنقول انه بيعت LSU فالمعنى انه بيعت LSU برضو.

16.1.3. At Change

- The router will send a partial triggered update

زي ما قلنا ان لو حصل تغيير بيتم ارسال LSU فيه التغيير اللي حصل في الشبكة فقط .. بعكس بروتوكول الـ RIP كان بيعت الـ Full Routing Table كل 30 ثانية.

16.2. OSPF Versions

IPv6 بيشتغل مع OSPFv2 و IPv4 بيشتغل مع OSPFv3

16.2. OSPF Network Types

في 3 انواع من الشبكات في شبكات الـ OSPF .. وهم Broadcast و Point-to-Point و بنفرق بينهم بناءا على نوع الـ Interface ونوع الـ Topology .. وفي نوع تالت كمان

- **النوع الاول:** مثل الـ **Point-to-Point Network** بيكون فيها راوترین فقط في نفس الـ Subnet

ومتوصلين بـ Serial Interface



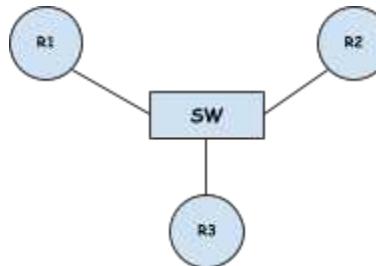
- النوع دا بيتمشى بنفس الخطوات اللي شرحناها فوق .. ومش بيعمل Election لـ DR

و BDR و Router

- والـ Priority بتاعت كل راوتر بتكون 0 .. حتى لو عدلتها >> بترجع تاني لـ 0
- الـ OSPF بيحدد نوع الـ شبكة بناءا على نوع الـ Interface .. وبالتالي لو لقى ان الـ Point-to-Point هيخللي الشبكة Serial

- **النوع الثاني:** اما الـ **Broadcast Network** بيكون فيها اكتر من راوتر في نفس الـ Subnet

ومتوصلين بأي نوع Ethernet.



- في شبكات الـ Broadcast بيكون في عدد كبير من الـ Routers وكل راوتر هيعمل Neighbor Relationship مع باقي الراوترات .. وبالتالي لو عندنا 10 راوتر مثلا هيبقى في OSPF Traffic كبير جدا هياثر على استقرار الشبكة وهيستهلك الـ CPU عشان يقدر يعالج الكم الكبير دا من الـ Traffic.

- وزي ما قلنا ان في نهاية عملية الـ Fully Neighbor State كل الروابط بيكون عندها نفس الـ LSDB .. فلو افترضنا ان عندنا شبكة OSPF مستقرة وفي راوتر جديد تم اضافته للشبكة، اللي هيحصل ان بعد ما يعدي الـ Two-Way Comm كل الروابط هتبعتلو نفس الـ DBD، وهنا في Traffic كبير هيستلمو ويعلمون Processing عالفاضي.
- عشان كدا بقى في حاجة لانتخاب DR و Designated Router عشان ينظموا العملية دي.
- الفكرة ان بيتم انتخاب DR و BDR وباقى الـ Routers بيكونو DRother .. وبعد كدا كل الـ Multicast Address: 224.0.0.6 على الـ DR والـ BDR على الـ DR والـ DRothers وبعدين الـ DR يشوف كل راوتر ناقصه ايه ويبعث على 224.0.0.5.
- الـ DR والـ BDR بيتم انتخابهم في مرحلة الـ ExStart .. والاتنين دول بيوصلو للـ Full DR والـ BDR بيتخابهم مع بعض ومع باقى الـ DRothers .. اما الـ DRothers يفضلو في الـ Adjacency State Two-Way-Comm بينهم وبين بعض.
- وبالتالي لو في راوتر جديد اتوصل بالشبكة «» هيوصل للـ Full State مع الـ DR والـ BDR فقط وبكدا هيتبادل معاهم الـ DBD وهيعرف LSR عشان يطلب الشبكات اللي ناقصاه والـ DR هيرد عليه بـ LSU.

• النوع الثالث: النوع الثالث اسمه Non-Broadcast Multi-Access Networks

- شبكات الـ NBMA بتشتغل بنظام الـ Point-to-Point يعني مش بتدعم الـ Broadcast وكل راوتر لازم يبقى فيه اتصال مباشر بينه وبين الرأوتر الثاني عشان يقدر يتواصل معاه.
- في شبكات الـ NBMA، زي X.25، ATM، Frame Relay، الـ OSPF بواجه مشكلة رئيسية وهي إن الشبكة مش بتدعم Broadcasts أو Multicast، وبالتالي مش هيقدر يكتشف الـ Neighbors تلقائياً باستخدام Hello Packets.
- عشان كدا لازم نحدد الـ Neighbors بشكل يدوى عشان الـ OSPF يقدر يرسل لهم التحديثات.

16.3. DR and BDR

انتخاب الـ DR والـ BDR يتم في مرحلة الـ ExStart ويتبع اختيارهم على حسب مجموعة من الخصائص:

- اول راوتر يقوم في خلال 40 ثانية
- الراوتر صاحب أعلى Priority .. وهو رقم من 0 الى 255 وب يكون 1 بشكل افتراضي
- الراوتر صاحب أعلى RID وهو رقم من 32 بت
 - الـ RID ممكن يكون Manually Configured
 - لو مش معمول بشكل Manual بياخد اعلى IP
 - ثم اعلى IP لاي Enabled Interface تاني لو مفيش
- الـ DR وظيفته انه يستقبل التحديثات من الاجهزه وهو اللي يوزعها

امتي بيحصل إعادة انتخاب او ازاي اخلي راوتر معين يكون الـ DR

- اعمل BDR للـ DR والـ Reload
- او اعمل Clear ip ospf process
- او غير الـ DR للـ DR والـ BDR وخليلها 0
- او اعمل BDR DR للـ DR والـ Shutdown/No shutdown

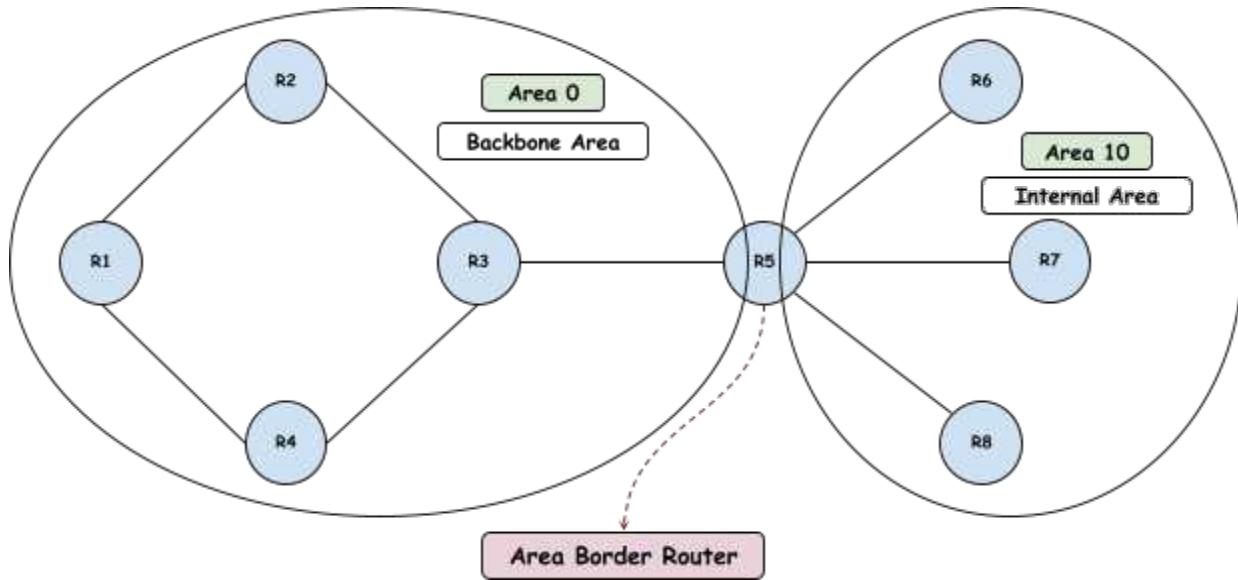
في الحالة دي لو عملتهم Shutdown هيحصل إعادة انتخاب للـ DR والـ BDR .. ولما اعملهم No Shutdown هيحصل تفاوض بين القدام والجداد وصاحب الاعلى Priority هو اللي هيحافظ على صلاحيته.

تفاصيل عن الـ Hello Packet :

- الـ Hello Interval بتكون 30 ثانية By Default
- الـ Dead Interval بتكون اربع اضعافها، يعني $120 = 30 * 4$ ثانية
- الـ Hello Interval بتتغير على حسب نوع الشبكة، وممكن اغييرها بشكل Manual
- لو غيرت الـ Hello Interval <> الـ Dead Interval هتتغير معها وهتبقى أربع اضعافها برضو .. أما لو غيرت الـ Hello Interval <> الـ Dead Interval مش هتناثر.

16.4. OSPF Areas

شبكة الـ OSPF بتعتمد على الـ Areas .. بس ايه لازمة الـ Areas دي !! .. عشان نفهمها ممكن نفترض ان دي شبكة خاصة باحد شركات الـ Service Provider

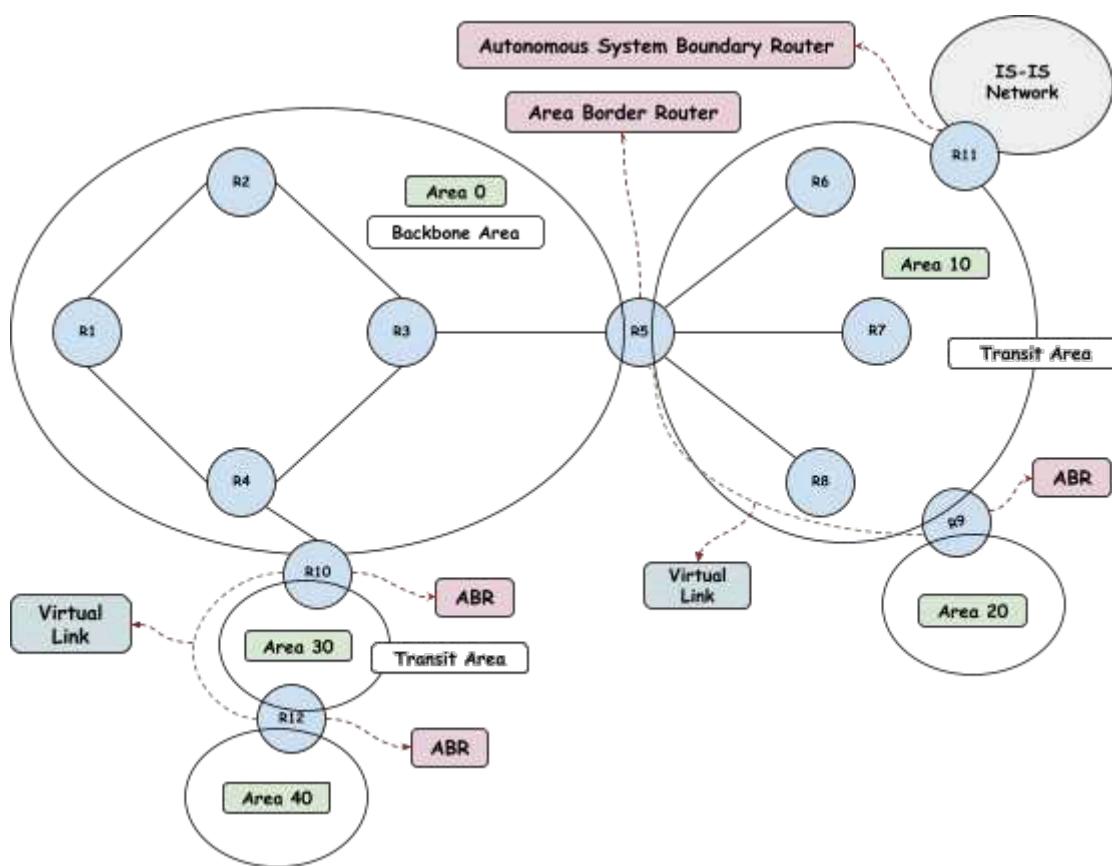


- لو في Router حصل عليه تغيير مثلاً «Generate LSA» ويعملها لكل الـ LSA ويعملها لكل الـ Routers كل Router يفتش عن طريق للوصول إلى الـ Area الذي في نفس الـ Area.
- وتأثير الـ LSA دي .. ان كل Router يطبق معادلة الـ Dijkstra عشان يحسب افضل مسار للشبكات اللي حصلها تغيير .. او على حسب التغيير ممكن يعيد كل الحسابات ودا هي عمل Load عالي على الـ CPU .. دا غير ان مع زيادة عدد الروابط والشبكات «الـ LSDB» هتزيد.
- عشان كدا افضل هو عمل تقسيم للشبكات الكبيرة الى Areas .. و CISCO بتنصح باستخدام من 50 الى 70 روتر في الـ Area الواحدة .. وبالتالي لو حصل تغيير «الـ LSA» هتتبعه للروابط اللي في نفس الـ Area فقط .. ودا هيقلل الـ Load.

كدا عرفنا ليه بنسخدم Areas مختلفة .. بس في مجموعة من المسميات اللي لازم نكون عارفينها في الـ :OSPF Areas

- دایما 0 (ممکن تكتب كدا 0.0.0.0) هي أول Area ويطلق عليها **Backbone Area** وكل الـ **Backbone Routers** اللي فيها يطلق عليهم **Routers**
- طيب ايه لازمة الـ Area بترتبط باقي الـ Areas بعض .. يعني عشان Area توصل لـ Area لازم يعدو على الـ **Backbone Area**
- أي Area متصلة بالـ Backbone Area واي Router فيها يطلق عليه **Internal Area** واي Internal Router
- ولو في Area متصلة بالـ Internal Area في الحالة دي هتبقى **Transit Area**

الرسم بيوضح المسميات بشكل أفضل

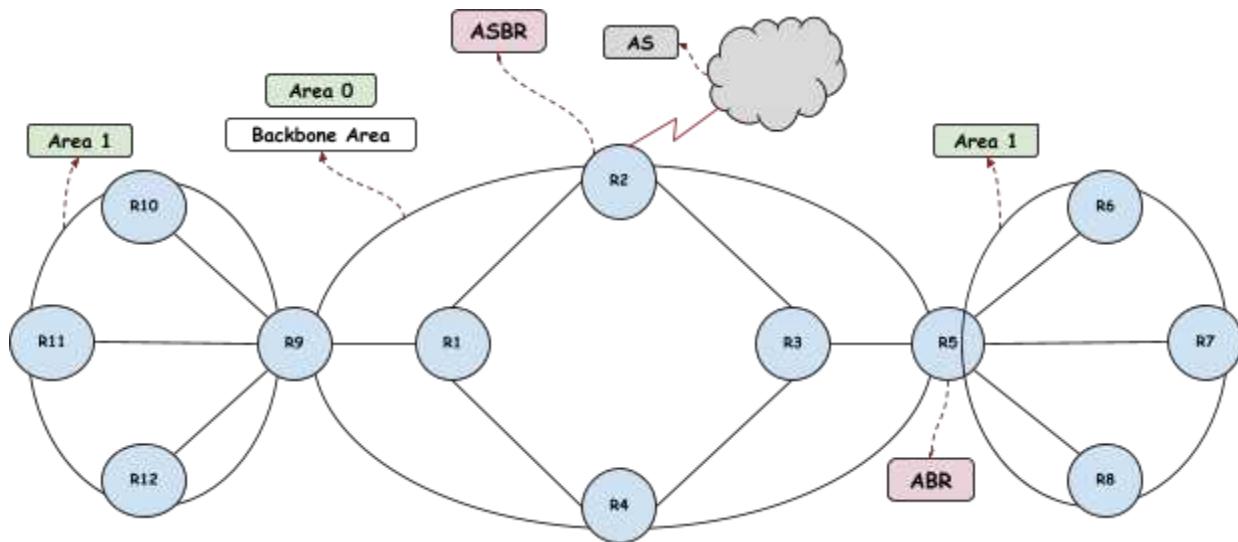


- الراوتر اللي يربط اتنين Area بـ Interface ولازم يكون فيه Area Border Router
- عشان كدا الـ ABR اللي بين A30 و A40 بنوصله بالـ Backbone Area عن طريق Virtual Link.

- الراوتر اللي يربط شبكة الـ OSPF بـ Autonomous System بـ OSPF او RIP او IS-IS او اي بروتوكول اسمه Autonomous System Boundary Router - ASBR

16.5. Area and LSA Types

- في 9 أنواع من الـ LSA ..
- سبعة منهم شغالين على IPv4
- واثنين شغالين على IPv6



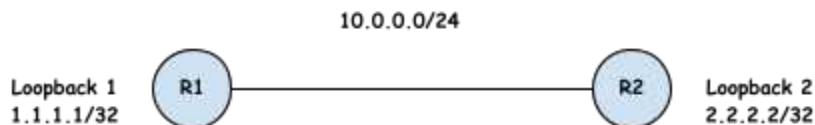
- اسمها اي Router يgenerate LSAs وهي الـ Type 1 LSA .نفس الـ Area

- اسمها DR اللي يبيعتها عشان يعرف باقي الـ Routers وهي الـ Network LSA وهي الـ Type 2 LSA . بنفسه ويعرفهم بباقي الـ Routers اللي في نفس الـ Area. وبتتبع في نفس الـ Area فقط.

- اسمها Type 3 LSA وهي Area Summary LSA تتبعت من Area الى Area تانية عن طريق الـ ABR .. يعني الـ ABR ي يعمل Generate Router Information Summary عن الـ ABR اللي عنده ويبعثها للـ Backbone Area وبعدين بتتبعت لباقي الـ Areas.
- اسمها Type 4 LSA وهي ASBR Summary LSA ويبعثها الـ ASBR لباقي الـ Routers اللي في كل الـ Areas عشان يعرفو مكانه.
- اسمها Type 5 LSA وهي Autonomous System LSA ويبعثها الـ ASBR والهدف منها نشر Static Routes جايه من برا الـ OSPF Autonomous System أو EIGRP Routes.
- اسمها Type 6 Multicast OSPF LSA : مش مستخدمة ومش مدعومة من CISCO اصلا.
- اسمها Type 7 :
بعد ما عرفنا انواع الـ LSA .. دلوقتي سهل نتعرف على انواع الـ Areas

- الـ Area 0 هي Backbone Area .
- الـ Area هي Standard Area عادية بتستقبل وتبعث كل أنواع الـ LSAs .
- الـ Stub Area هي الـ Areas اللي بتمنع T5 LSA .. يعني مفيهاش Routes خارجية جايه من برا الـ OSPF Autonomous System .
- الـ Totally Stopped Area هي الـ Area اللي بتمنع T3, T4, T5 LSAs يعني تقريباً أي حاجة مش جوه Area نفسها بتترفض .
- اختصار لـ Not-So-Stubby Area ودي نسخة معدلة من Stub Area بتمنع Type 5 LSA برضه، بس بتسمح بإرسال الـ Routes الخارجية عن طريق إرسال الـ Type 7 LSA .. وفكرة الـ Type 7 انها بتعمل Cover لـ Backbone Area اللي واصل بالـ ABR اللي بدوره هيشيل الـ Cover ويبعثها لباقي الـ Areas .

16.6. OSPF Lab-1



اول حاجة المفروض نعمل لـ Loopback Interface Configuration

```
R1(config)# int loopback 1
R1(config-if)# ip add 1.1.1.1 255.255.255.255
```

```
R2(config)# int loopback 2
R2(config-if)# ip add 2.2.2.2 255.255.255.255
```

وبعدين نعمل لـ IPs Configuration

```
R1(config)# int f0/0
R1(config-if)# ip add 10.0.0.1 255.255.255.0
```

```
R2(config)# int f0/0
R2(config-if)# ip add 10.0.0.2 255.255.255.0
```

لتشغيل الـ OSPF بنتحتاج نحدد Process ID باستخدام الأمر **router ospf 1** وهو رقم Local Significant .. يعني تأثيره محلي على الراوتر الحالي فقط وهو عبارة عن Instance مختلفة من الـ OSPF وبالتالي لو عملت لـ OSPF بـ Process ID مختلف على راوترتين .. الاتنين هيشوفو بعض عادي .. أما لو عملت اتنين Process ID على نفس الراوتر <> هيفي كل Instance لها Database منفصلة عن الثانية .. ودا مفيد في حالة الـ SP عشان يربط بين اكتر من Customer وبالتالي كل شركة يكون لها Database منفصلة.

```
R1(config)# router ospf 1
R1(config-router)# network 10.0.0.1 0.0.0.255 area 0
```

```
R2(config)# router ospf 1
R2(config-router)# network 10.0.0.2 0.0.0.255 area 0
```

ملحوظة: لو عدد الـ Router Crash لـ Routes .. ممكن يحصل لـ Router او الـ Process ID نفسها.

اما الامر **network 10.0.0.1 0.0.0.255 area 0** بيعمل 4 مهام

- بيعمل Network لـ Advertise اللي حدتها
- بيعمل Network Range على الـ Interfaces اللي في نفس الـ OSPF على نفس الـ Network المحدد، وبالتالي الـ Hello Packets تتبعت من خلال الـ Interfaces دي
- بعد الـ Network هنالاحظ وجود 0.0.0.255 ودا اسمه Wild Card وهو مقلوب الـ Subnet Mask
- لحساب الـ Wildcard .. حول الـ Subnet Mask الى Binary ثم اعكس الاصفار والواحدات ورجعوا لـ Decimal تاني .. او بسهولة اطرح الـ Subnet Mask من 255.255.255.255
- تحديد الـ Area

ملحوظة: مصطلح ISP يستخدم لربط افرع الشركات .. أما ISP لتقديم خدمة الـ Internet

الـ Wild Card بيأدي نفس وظيفة الـ Subnet Mask عشان يحدد الـ Matched Interfaces فمثلا لو عايز اشغل الـ OSPF على كل الـ Interfaces، ممكن استخدم الامر التالي:

```
R1(config-router)# network 0.0.0.0 255.255.255.255
```

بعد تحديد الـ Area وعمل Syslog على كل راوتر .. هتظهر Interface No shutdown لـ Matched Interfaces فيها الـ Neighbor RID وحالته Full State يعني انهم بقو From Loading to full ووصلو لـ Neighbors

```
R1# %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on FastEthernet0/0 from LOADING to FULL, Loading Done
```

```
R2# %OSPF-5-ADJCHG: Process 1, Nbr 1.1.1.1 on FastEthernet0/0 from LOADING to FULL, Loading Done
```

في 3 انواع من الجداول في الـ OSPF

- جدول الـ Neighbors .. وممكن اعرضه باستخدام الامر:

```
R1# show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
2.2.2.2	1	FULL/BDR	00:00:32	10.0.0.2	FastEthernet0/0

- OSPF Database Table

```
R1# show ip ospf database
```

OSPF Router with ID (1.1.1.1) (Process ID 1)

Router Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum	Link count
1.1.1.1	1.1.1.1	453	0x80000002	0x006AA9	1
2.2.2.2	2.2.2.2	454	0x80000002	0x002CDE	1

Net Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum
10.0.0.1	1.1.1.1	453	0x80000001	0x007BA3

هنا يظهر الـ LSAs الموجودة على الـ Router .. وزي ما هو واضح .. في Router LSA اللي عملها كل

راوتر وتبادلوها .. وفي Network LSA الخاصة بالـ DR

- Routing Table

```
R1# show ip route
```

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area

N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2

E1 - OSPF external type 1, E2 - OSPF external type 2

i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2

ia - IS-IS inter area, * - candidate default, U - per-user static

route

o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP

```
+ - replicated route, % - next hop override
```

```
Gateway of last resort is not set
```

```
    1.0.0.0/32 is subnetted, 1 subnets
C      1.1.1.1 is directly connected, Loopback1
10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C      10.0.0.0/24 is directly connected, FastEthernet0/0
L      10.0.0.1/32 is directly connected, FastEthernet0/0
R1#
```

في الـ Route الـ Route اللي في نفس الـ Area بيظهر معاه حرف O

الـ Route اللي معاه حرف IA معناه ان الـ Route جاي من Area تانية

OSPF Troubleshoot

لو ظهر Syslog بالشكل دا

```
R1(config-if)#
```

```
*Feb 4 04:25:53.751: %OSPF-5-ADJCHG: Process 1, Nbr 2.2.2.2 on FastEthernet0/0
from FULL to DOWN, Neighbor Down: Interface down or detached
```

- واضح من الـ Log ان في مشكلة في الـ Neighborship مع الراوتر صاحب الـ RID:2.2.2.2 وهو R2
- وبالتالي ممكن نستنتج ان في عدم توافق في احد شروط الـ Neighbor Relationship .. واللي هيا Directly Connected
- لازم يكون لهم نفس الـ Subnet Mask ونفس الـ Hello والـ Dead Interval ويكونو Connected
- وبالتالي عشان نعرف المشكلة بالضبط لازم نستخدم الـ Debugging .. واللي في منه انواع كتير .. منهم Hello Packet للـ Debug

```

R1# debug ip ospf hello
OSPF hello debugging is on
R1#
*Feb 4 04:34:57.395: OSPF-1 HELLO Fa0/0: Send hello to 224.0.0.5 area 0 from
10.0.0.1
*Feb 4 04:34:58.071: OSPF-1 HELLO Fa0/0: Rcv hello from 2.2.2.2 area 0 10.0.0.2
*Feb 4 04:34:58.071: OSPF-1 HELLO Fa0/0: Mismatched hello parameters from
10.0.0.2
*Feb 4 04:34:58.071: OSPF-1 HELLO Fa0/0: Dead R 40 C 40, Hello R 10 C 10 Mask R
255.255.255.0 C 255.255.0.0
R1# no debug ip ospf hello
OSPF hello debugging is off

```

ممكن نعمل الغاء لل Debugging بمجرد ما نستلم Hello .. لأن الـ Debugging على الـ CPU

هنالاحظ فالاول ان في الـ Hello Parameters Mismatch فعلاً في الـ

- معناها إن الـ Dead Interval متطابق (40 ثانية)، ودي مش سبب المشكلة.

- معناها إن الـ Hello Interval متطابق (10 ثواني)، ودي برضو مش المشكلة.

- الراوتر الثاني (R2) بيستخدم الماسك ده.

- الراوتر R1 بيستخدم ماسك مختلف.

وبكدا عرفنا السبب الرئيسي لعدم تكوين علاقة الـ OSPF Neighbor Adjacency .

لتغيير الـ Hello Interval .. انزل تحت الـ Interface وطبق الأمر التالي

```
R1(config-if)# ip ospf hello-interval 20
```

عشان ارجعها للـ Default

```
R1(config-if)# no ip ospf hello-interval 20
```

ملحوظة: ممكن حد يفكرة بتغيير الـ Hello Interval الى ثانية واحدة مثلاً عشان يخليل الشبكة سريعة .. بس تأثير انه هيعمل Load على الـ CPU .. لأن كل ثانية هيستلم Hello من كل الـ Routers ولازم يعملها Future .. ولكن في الـ Future اسمه BFD بيساعد على تسريع الشبكة Processing

لتحفيير الـ Priority

```
R1(config-if)# ip ospf priority 5
```

لتحفيير الـ RID لراوتر معين

```
R1(config-router)# router-id 5.5.5.5
```

16.6.1. OSPF Authentication

باستخدام الـ Authentication .. الراوترات مش بتقبل اي رسائل Hello او اي Updates الا لو جاية من Routers بتسخدم نفس الـ Password اللي حددته.

في اتنين Modes من الـ Authentication في الـ OSPF

- Null - Type 0 - يعني مفيش مصادقة
- Cleartext - Type 1 - ip ospf authentication
- MD5 - Type 2 - ip ospf authentication message-digest

لتطبيق Clear-text Authentication على مستوى الـ Interface .. اكتب الامر

```
R1(config)# int f0/0
R1(config-if)# ip ospf authentication
R1(config-if)# ip ospf authentication-key 12599
```

الـ Auth key آخره تمانية 8 .. ولو كتبت اكتر من 8، هيختار اول 8 فقط

لتطبيق MD5 Authentication .. اكتب الامر

```
R1(config-if)# ip ospf authentication message-digest
R1(config-if)# ip ospf authentication-key 1 MD5 123
```

لعمل Troubleshoot .. استخدم الامر debug adjacency

16.6.2. OSPF Lab-2

How to calculate OSPF Cost (dijkstra algorithm)

OSPF cost = Reference Bandwidth / Interface Bandwidth, Where Ref.. BW is 10^8

- For Ethernet: $BW = 10 \text{ Mbps} = 10 * 10^6 \text{ bps}$
 $Cost = 10^8 / 10 * 10^6 = 10$
- For FastEthernet: $Cost = 10^8 / 100 * 10^6 = 1$
- For GigEthernet: $Cost = 10^8 / 1000 * 10^6 = 1$

المفروض في الـ Cost الـ GigEth يطلع 0.1 بس الـ OSPF مش بيفهم الكسور .. فبيقرب لأقرب رقم

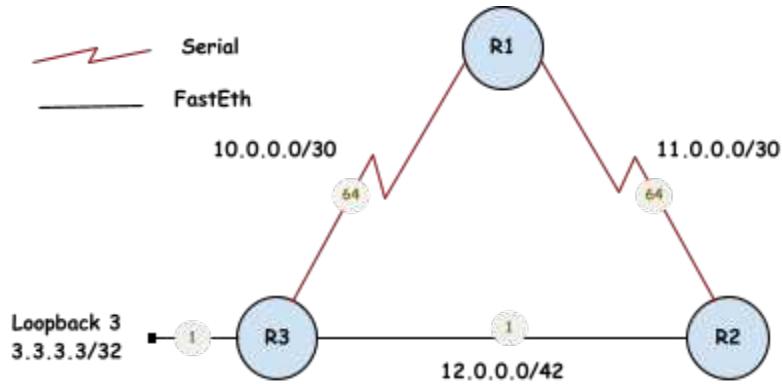
صحيح .. وبالتالي الـ OSPF مش بيفرق بين الـ GigEth والـ FastEth .. ومع ذلك اقدر اغير القيمة دي

- For Serial: $Cost = 10^8 / 1.54 * 10^6 = 64$

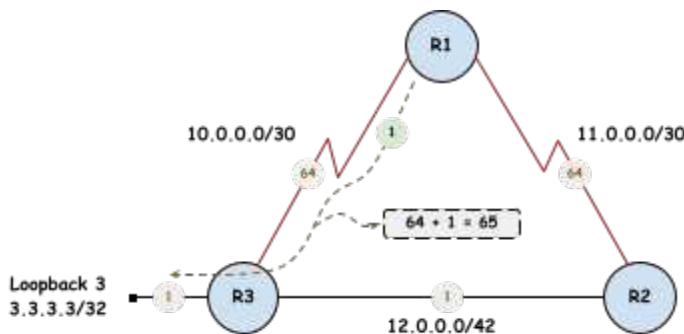
للتفريق بين الـ FastEth والـ GigEth ممكن اغير الـ Reference Bandwidth بتاع المعادلة .. فبدل 10^8 ممكن اخليها 10^9 وكدا الـ Cost بتاع الـ FastEth هيبقى 10.

تنويه: لو هتغير قيمة الـ Reference Bandwidth .. راعي ان انتا متزودوش بشكل كبير لأن فالآخر الـ OSPF بيجمع الـ Costs عشان يحسب الـ Metric واللي اخره 4 مليار وشوية

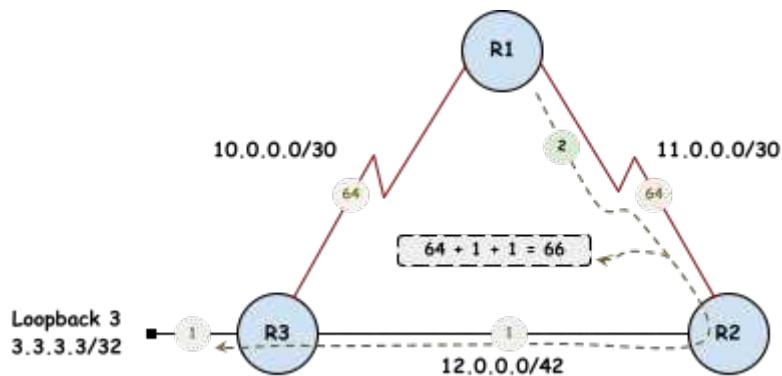
لحساب الـ Cost من R1 الى 3.3.3.3/32 .. هنلاحظ وجود مسارات :



- أول مسار: الـ Traffic يخرج من R1 الى R3 ثم إلى الشبكة •
الـ Cost الخاص بالـ FastEth = 1 وبعدين الـ Serial Interface .. يبقى
المحصلة 65



- ثاني مسار: الـ Traffic يخرج من R2 الى R1 ثم الى R3 .. وبالتالي الـ Cost هيختار اول مسار لانه اقل



قلنا قبل كدا ان الـ Bandwidth هو أقصى سرعة لمرور البيانات داخل الشبكة .. ولكن التعريف دا مش صحيح لأن الـ BW قيمة Configurable، يعني اقدر اغييرها بالنسبة لاي Interface. فمثلا لو اشتريت سرعة 10 Mbps لـ Path1 وسرعة 40 Mbps لـ Path2 برضو الـ OSPF هيختار مسار 1.

لو كان الـ OSPF يعتمد على السرعات الحقيقية لمرور البيانات في الشبكة .. كان ممكن الروابط تتحرق، لأن السرعة الحقيقية بتتغير باستمرار، وبالتالي هيفضل يشغل معادلة الـ SPF لحساب المسار الأفضل.

CISCO بتسمح بتغيير قيمة الـ Cost لأي مسار عشان اخلي البروتوكول يختار المسار اللي انا عاوزو .. والعملية دي اسمها Traffic Engineering.

لتشغيل الـ OSPF على Interface معين مباشرة من غير ما احتاج احدد الـ Interfaces عن طريق الـ Wild Card Network .. استخدم أمر:

```
R1(config)# int f0/0
R1(config-if)# ip ospf 1 area 0
```

where "1" is the Process ID

لتغيير الـ Cost لأي Interface

```
R1(config)# int f0/0
R1(config-if)# ip ospf cost 30
```

لتغيير الـ Reference Bandwidth لمعادلة الـ SPF

```
R1(config)# router ospf 1
R1(config-router)# auto-cost reference-bandwidth 1000 [1000 in Mbps]
```

تغيير الـ Reference-bandwidth على الـ OSPF على الـ Router Relationship مش بيأثر على الـ EIGRP لازم اطبقها على كل الـ Routers.

16.7. Default Information Originate

في الـ OSPF ممكن نعمل الأمر `Default Route` لـ `Configure` باستخدام الأمر `default-information originate`، والأمر ممكن استخدامه لو عندي اكتر من `Branches` متصلين ببعض عن طريق الـ `SP`، وفي واحد منهم "الفرع الرئيسي" مثل واصل بالـ `Internet` .. فبدلاً من توصيل كل فرع بالـ `Internet` او بدل ما اعمل على كل فرع بشكل `Manual` .. ممكن اخلي الراوتر اللي في الفرع الرئيسي يعمل `Default Route` لـ `Default Route Advertise`.

وفي بعض الـ Options مع الامر دا

```
R1(config-router)# default-information originate ?
  always      Always advertise default route
  metric      OSPF default metric
  metric-type OSPF metric type for default routes
  route-map   Route-map reference
<cr>
```

واهم `default-information originate` هو `always` .. لأن لو استخدمنا الامر `always` لوحدو،
هي عمل `Default Route Advertise` بتاع الراوتر بس لازم يكون في `Default Route` موجود في الـ
Routing Table حتى لو مفيش `Default Route` اما مع استخدام `always` هي عمل `Advertise` لنفسه كـ `Default Route` في الـ Routing Table.

وهي ظهرت عند باقي الـ Routers بالشكل دا

```
R2# show ip ospf database | begin Type-5
    Type-5 AS External Link States
  Link ID        ADV Router       Age        Seq#      Checksum Tag
  0.0.0.0        10.0.0.1        59        0x80000001  0x008D64 1
```

```
R2# show ip route ospf
0*E2 0.0.0.0/0 [110/1] via 10.0.0.1, 00:00:24, FastEthernet0/0
```

ونفس الأمر مدعوم في بروتوكول الـ RIP بس من غير "always" لأن مش لازم يكون عندي

لما تشغّل OSPF على Loopback Interface، يتم الإعلان عنها بـ /32 Prefix بغض النظر عن الـ Mask الأصلي، لأنّه يتعامل مع الـ Loopback كـ Host Route (يعني عنوان IP فردي مش شبّكة كاملة).

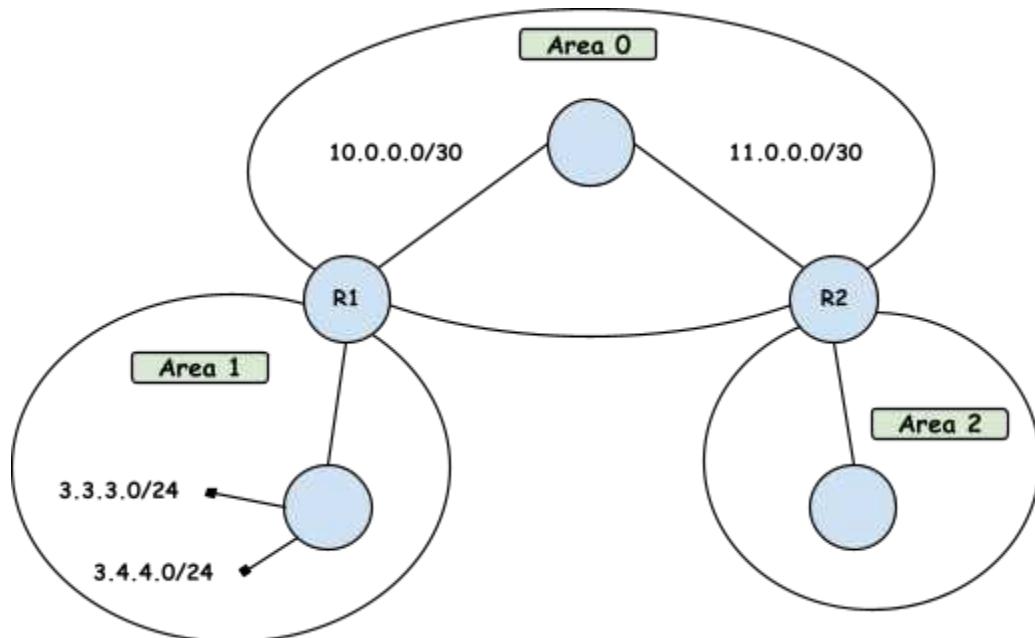
تقدر تغيير السلوك الافتراضي عن طريق أمر:

```
R1(config)# interface loopback 0
R1(config-if)# ip ospf network point-to-point
```

16.8. Interarea Route Summarization

الـ OSPF ممكّن يعمل Summarization بس مش في نفس الـ Area .. يعني لازم اعملو Configuration على الـ ABR Router او الـ ASBR .. عشان يعمل Summarization للـ Routes الموجودة في معينة Area وبيعتها الى Area تانية بـ Type 3 او 5 Type LSA. بعكس بروتوكول RIP او EIGRP اللي ممكّن يطبقها على أي Router.

ايه تأثير الـ Summarization



لو معمليش Summarization في الشبكة اللي فوق .. كل Prefix هيكون لها LSA وبالتالي لو شبكة 3.3.3.0/24 وقعت مثلا .. R1 وهو ABR هيعمل Flood لـ Network LSA وتهتوصل لكل الـ Routers في الشبكة وبالتالي هيحصل Re-run لعملية حساب الـ Cost على كل Router.

اما لو عملت Summarization ممكن اخلي 3.4.4.0/24 و 3.3.3.0/24 يتعملاهم الى 3.0.0.0/8 وبالتالي لو حصل Fail في احد الشبكات في Area 1 مفيش تأثير هيحصل على باقي الـ Areas لأن معندهم معلومة عن الشبكة المحددة اللي حصلها Failure .. هما عندهم الا Summarized Network فقط .. عشان كدا مش هيحصل تأثير الا لو الشبكتين وقعوا.

ملاحظات

- عشان نعمل Summary .. لازم يكون في شبكة واحدة على الأقل موجودة في الا Summary Range
- الـ Summary Route بيأخذ Cost اقل شبكة في الا Summary Range

لعمل Summarization للشبكتين 3.3.3.0/24 و 3.4.4.0/24

```
R1(config)# router ospf 1
R1(config-if)# area 1 range 3.0.0.0 255.0.0.0
```

لو في راوتر متوصل بـ Enduser .. ممكن يعمل Attack على الا Router ويبيع ويستلم Hello، ودا هيأثر على الشبكة .. عشان كدا ممكن نحول الا Interfaces المتوصلة بـ Endusers او بـ LAN Network الى Passive Interface عشان مبيعدتش ولا يستلم Hello عليها .. ودا ممكن عن طريق الامر:

```
R1(config-router)# passive-interface loopb 0
```

تطبيقاتها على كل الا Interfaces

```
R1(config-router)# passive-interface default
```

17. Serial Interfaces

ال WAN Interface هو نوع من الـ **Interfaces** اللي بتسخدم في شبكات الـ **WAN**، ويسمح بتوصيل الشبكات المحلية (LANs) ببعضها عن طريق مزود خدمة الإنترنت (ISP) أو عن طريق تقنيات تانية زي الـ **Leased Lines**. بيستغل من خلال إرسال واستقبال البيانات بسرعات مختلفة حسب نوع الخدمة المتاحة من الـ **ISP**.

في نوعين من الـ **Wan Technology**:

- الـ **Public WAN**: يستخدم لربط العميل بالـ **Internet**.
- الـ **Private WAN**: بيستخدم لربط افرع الشركات ببعضها عن طريق الـ **ISP**. وفيه اكتر طريقة لربط الفروع أو الشبكات المحلية (LANs)، ومنها تقنية **Leased Line Technology** اللي بتحتاج استخدام **Serial Interface**.



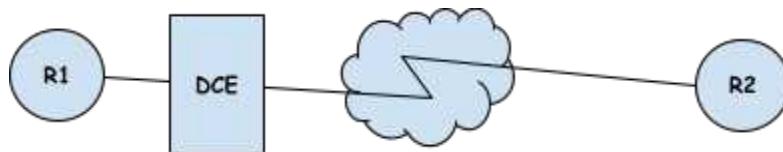
WAN و DCE و DTE في شبكات الـ

الراوتر بيسمى **DTE - Data Terminal Equipment**، يعني بيبعد البيانات بسرعة الـ **Interface** الخاص بي. وأقصى سرعة الـ **Serial Interface** بتوصيل لـ 45 ميجابت في الثانية، لكن مع تطور تقنيات **WAN**، استخدام الـ **Serial Interfaces** قل جدا.

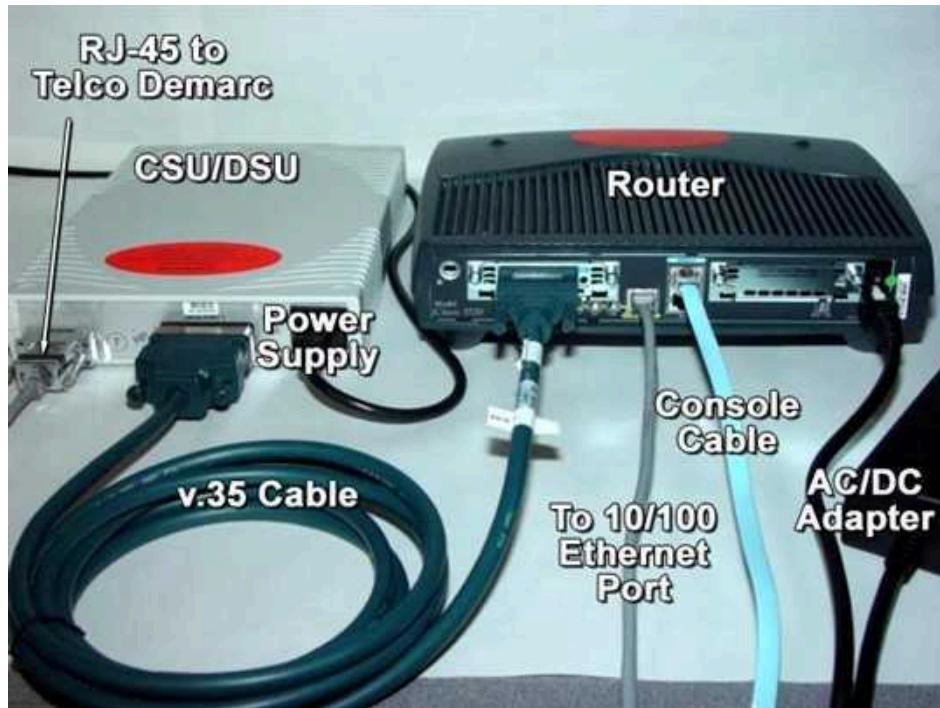
لو العميل متعاقد مع مزود الخدمة على سرعة أقل من الحد الأقصى، لازم نوصل الراوتر بجهاز اسمه **DCE Data Communication Equipment**، والجهاز ده هو اللي بيتحكم في الاتصال وسرعته.

طريقة عمل الـ DCE

الـ Service Provider يقسم الثانية الى مجموعة من الاجزاء .. ممكن نفترض انه بيقسمها الى 8 اجزاء مثل الـ حسب السرعة اللي العميل متعاقد عليها بيتحكمو في عدد الأجزاء من الثانية اللي البوت بيفتح فيها.



جهاز الـ DCE ممكن يكون Analog زي الـ CSU/DSU او Digital Modem زي الـ Digital .. حاليا بقا الـ DCE والـ DSL Modem موجودين في جهاز واحد زي الـ DTE.



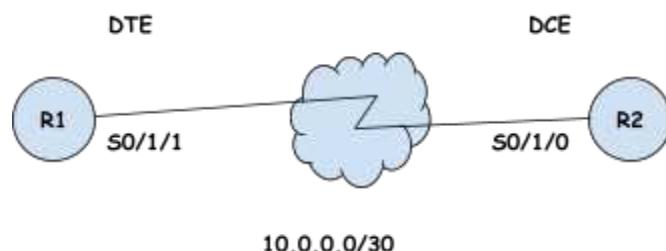
صورة قديمة لجهاز الـ Modem متصل بالـ CSU/DSU

- الـ Serial Cable في منه انواع كتير .. وشهرها هو v.35
- الـ Serial Interface لازم احط عليه IP وبعدين اعمله No shutdown .Clocking by default .. بس في انواع بيبقى عليها Clocking عشان يشتغل

عشان نطبق ال LAB هنوصل اتنين Router ببعض ونوصلهم بـ Serial Cable .. وفي الواقع ال Cable دا بيقيا عبارة عن طرفيين، واحد DTE والطرف الثاني DCE.



أحد الرواترات هي تعامل معاملة لا DCE وهي تحكم في الرواتر الثاني



R1

```
R1(config)#int s0/1/1
R1(config-if)#ip add 10.0.0.1 255.255.255.252
R1(config-if)#no shut
R1(config-if)#
R1(config-if)#
*Jan  1 00:34:20.211: %LINK-3-UPDOWN: Interface Serial0/1/1, changed state to do
wn
R1(config-if)#

```

R1(config-if)#do show ip int b	IP-Address	OK?	Method	Status	Prot
Interface ocol	unassigned	YES	unset	administratively down	down
FastEthernet0/0	unassigned	YES	unset	administratively down	down
FastEthernet0/1	unassigned	YES	unset	administratively down	down
Serial0/1/0	unassigned	YES	unset	administratively down	down
Serial0/1/1	10.0.0.1	YES	manual	down	down

حالة المنفذ Down لأن الطرف المقابل لسا مشتغلش

R2

Interface	IP-Address	OK? Method Status	Prot
Serial0/1/0	10.0.0.2	YES manual up	up
FastEthernet0/0	unassigned	YES unset administratively down	down
FastEthernet0/1	unassigned	YES unset administratively down	down

بعد ما حطيت الـ IP وعملت `No shutdown` في بعض الحالات ممكن يظهر ان حالة المنفذ `Up Down` بسبب اني معمليش `Clocking by default` بس الموديل دا فيه

عرض حالة الـ Interface هل هو DCE ولا DTE عن طريق الامر

```
R2#show controllers s0/1/0
Interface Serial0/1/0
Hardware is GT96K
DCE V.35, clock rate 125000
idb at 0x6418CA8C, driver data structure at 0x6418EE30
wic_info 0x6418F454
Physical Port 0, SCC Num 0
```

لاحظ ان نوع البورت `DCE` ونوع الكابل `V.35` والـ `Clock rate` عبارة عن `bps 125000`

```
R1#show controllers s0/1/1
Interface Serial0/1/1
Hardware is GT96K
DTE V.35 TX and RX clocks detected.
idb at 0x640CF948, driver data structure at 0x640D1CEC
wic_info 0x640D2310
Physical Port 2, SCC Num 2
```

على R1 ظهر ان نوع البورت `DTE` ونوع الكابل `V.35` واكتف ان في حد بيتحكم في الـ `Clocking`

لتفغير الـ Clock rate على الـ DCE Router

```
R2(config)#int s0/1/0
R2(config-if)#clock rate ?
      Speed (bits per second)
1200
2400
4800
9600
14400
19200
28800
32000
38400
48000
56000
57600
64000
72000
115200
125000
128000
<300-4000000>      Choose clockrate from list above
```

اللي بيتحكم في السرعات دي هو الـ ISP وبنفرق في تكلفة الخدمة .. ويقدر يكتشف اي تلاعب لو حصل من المستخدم

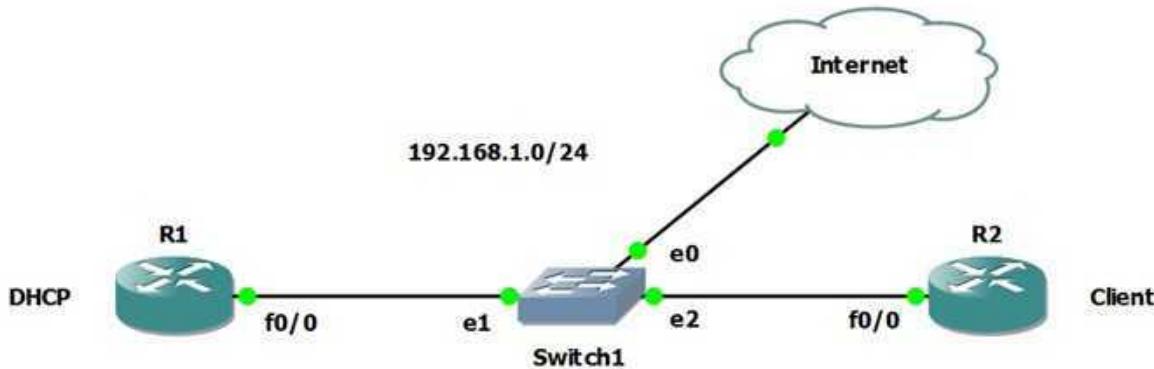
لو حاولت اغير الـ Clocking على الـ DTE Router يعني مش هيظهر وممكن يظهر Silent Error .. هيحصل على بعض الموديلات .. مثلا هنا مظهرش Running Config وفي الـ Error التعديل متمنش.

```
R1(config-if)#clock rate 72000
R1(config-if)#end
R1#
R1#
*Jan  1 00:40:39.603: %SYS-5-CONFIG_I: Configured from console by console
R1#
R1#show run int s0/1/1
Building configuration...

Current configuration : 66 bytes
!
interface Serial0/1/1
  ip address 10.0.0.1 255.255.255.252
end
```

لو عملت Ping بين الراوترین هنلاحظ أن كل الـ Ping Packets وصلت، في حين ان لو موصلهم بـ Ethernet هنلاقي غالبا اول Packet مش بتوصل بسبب ان الراوتر بيجب الـ MAC عن طريق الـ ARP .. اما الـ Serial Interface .MAC Address مفهوش

18. DHCPv4



الهدف من الدرس دي إعداد DHCPv4 Server على R1 لتوزيع عناوين IP بشكل Dynamic على الأجهزة اللي في الشبكة، بما فيهم الراوتر R2 اللي هنعتبرو DHCP Client. إل المتصولة به Switch1. لـ Cloud Client بـ Simulate ان الشبكة متصلة بال Internet عن طريق الـ .Cloud

DHCP Configuration

أول حاجة لازم إل DHCP يكون واحد Static IP

```
R1(config)# interface fastEthernet 0/0
R1(config-if)# ip address 192.168.1.1 255.255.255.0
R1(config-if)# no shutdown
```

خدمة DHCPv4 بتكون مفعولة By Default على أجهزة CISCO .. ولاغاء تفعيلها او لاعادة تفعيلها

```
R1(config)# service dhcp
```

الـ DHCP عبارة عن Pool فيها مجموعة او Range من الـ IP's .. وبالتالي لازم اعملExclude لمجموعة من الـ IP's الاستثنائية زي الـ Servers والـ Printers اللي لازم تاخذ IP ثابت.

```
R1(config)# ip dhcp excluded-address ?
A.B.C.D Low IP address

R1(config)# ip dhcp excluded-address 192.168.1.10 ?
A.B.C.D High IP address
<cr>

R1(config)# ip dhcp excluded-address 192.168.1.10 192.168.1.15
```

ثالث خطوة: إنشاء DHCP Pool باسم LAN_POOL وتحديد الشبكة اللي هيتم توزيع الـ IP's منها .. فايدة تحديد الاسم هواني ممكن اعمل اكتر من Pool وكل Pool لها اسم وشبكة معينة عشان الراوتر يقدر يوزع الـ IP المناسب للجهاز اللي ارسل .DHCP Request

```
R1(config)# ip dhcp pool LAN_POOL
R1(dhcp-config)# network 192.168.1.0 255.255.255.0
R1(dhcp-config)# dns-server 8.8.8.8 4.2.2.2
R1(dhcp-config)# default-router 192.168.1.1
```

- ممكن تستبدل الـ Subnet Mask وكتبه عن طريق الـ Prefix .. مثلـ 24 /24
- بعد إنشاء الـ Pool .. ممكن نحدد اكتر من Option لتوزيعه مع الـ IP زي الـ DNS والـ Gateway وغيرها
- ممكن احدد اكتر من DNS لحد 8
- الأمر Gateway هو الـ Default Router

لتغيير فترة الـ Lease Period والتي تكون يوم By Default على اجهزة Cisco

```
R1(config)# lease ?
<0-365> Days
infinite Infinite lease
```

لإضافة Dynamic IP على الراوتر

```
R2(config)# int f0/0
R2(config-if)# ip add dhcp
*Mar 1 00:19:26.503: %DHCP-6-ADDRESS_ASSIGN: Interface FastEthernet0/0
assigned DHCP address 192.168.1.2, mask 255.255.255.0, hostname R2
```

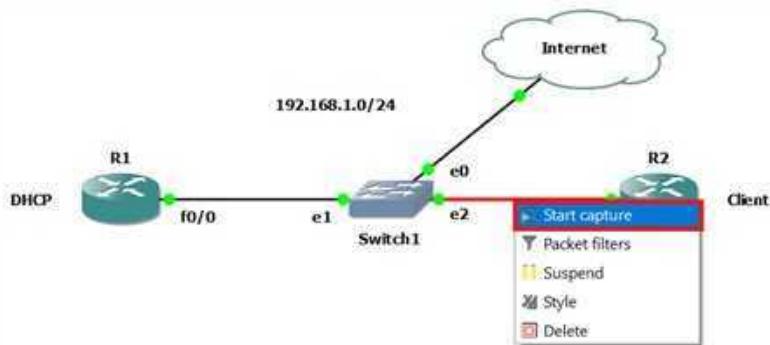
لإضافة Dynamic IP على Virtual PC استخدم الأمر PC# ip dhcp

من الـ Options كمان اللي اقدر اخلي الـ DHCP Server يوزعها، هي الـ Domain Name عن طريق امر

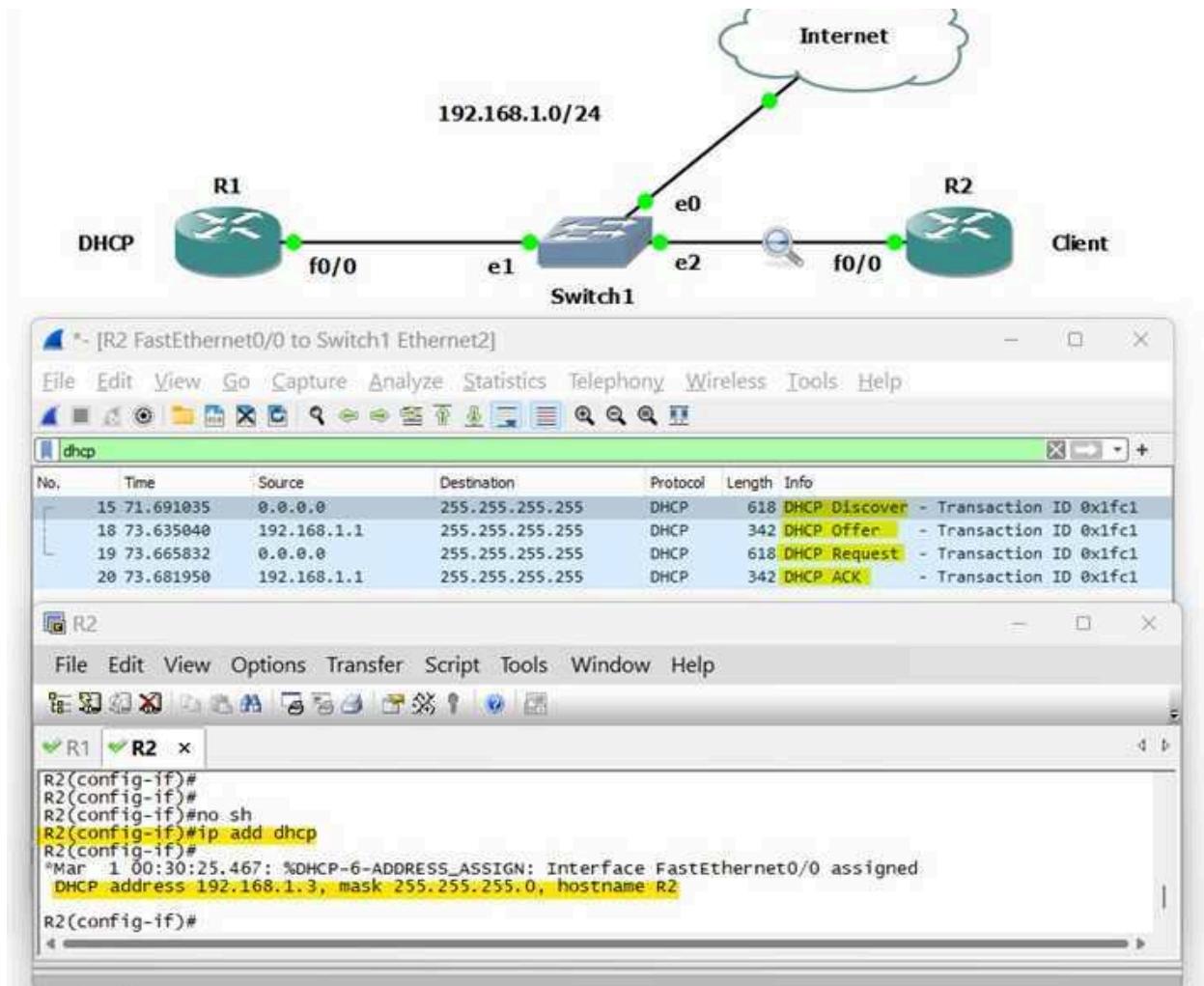
```
R1(dhcp-config)# domain-name ?
WORD Domain name
R1(dhcp-config)# domain-name tshoot.com
```

- دلوقتي لما أي جهاز يحصل على IP من الراوتر، هيتقم ضبط DNS Suffix بقائه بحيث يكون .tshoot.com
- يعني لو حاولت توصل لجهاز اسمه PC1 داخل الشبكة، السيستم هيتترجم الاسم تلقائياً لـ PC1.tshoot.com موجود ويعامل مع الدومين ده.

عرض عملية الـ DORA باستخدام Wireshark



ممكن تكتب في خانة الـ Filtering اللي فوق DHCP او Bootp لتسهيل البحث، لأن بيقى في Traffic كتير جدا



نبذة تاريخية الأول عن التطورات اللي حصلت على ما وصلنا لبروتوكول الـ DHCP

- في البداية، كانت أول محاولة لتوزيع عناوين IP بشكل ديناميكي باستخدام ما يُعرف بـ RARP Reverse Address Resolution Protocol .. وهنا الـ Server بيعتمد على بروتوكول Admin ي يعمل Database بعناوين الـ MAC الخاصة بالأجهزة والـ IP المرتبط بكل MAC .. وبالتالي لو جهاز تحتاج IP بيستخدم بروتوكول RARP عشان يأخذ IP من الـ Server
- مع الوقت، ظهر بروتوكول BOOTP كحل أكثر تطورا، ولكنه كان محدود بقدراته على توزيع IP فقط، ولا يدعم أي خيارات إضافية.
- وأخيرا، تم تطوير بروتوكول DHCP ليكون نسخة موسعة ومحسنة من BOOTP. والبروتوكول دا بجانب توزيع عناوين IP و Subnet Mask و Address Options زي Gateway، DNS Servers، Subnet Mask وغيرها.

عرض معلومات عن الأجهزة اللي خدت Dynamic IP من الـ DHCP Server

```
R1# show ip dhcp binding
R1# show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address          Client-ID/          Lease expiration      Type
                  Hardware address/
                  User name
192.168.1.3        0063.6973.636f.2d63.    Mar 02 2002 12:39 AM  Automatic
                    3230.332e.3065.3030.
                    2e30.3030.302d.4661.
                    302f.30
```

الـ MAC Address أو الـ Hardware Address أو الـ Client-ID الخاص بالجهاز

لـ DHCP Binding Table

```
R1(config)# clear ip dhcp binding *
```

او لـ IP معين

```
R1(config)# clear ip dhcp binding <ip address>
```

18.1. Reservation

ممكن استخدم الـ Reservation هشان لو عندي Printer مثلًا مش بتدعم اني اضيف لها Static IP فممكن احجزلها IP عن طريق الـ MAC Address الخاص بيها .. وبالتالي كل مرة الجهاز ده يطلب IP من الـ DHCP Server . هياخد نفس الـ IP بدون تغيير.

لمعرفة الـ MAC الخاص بالأجهزة المتصلة بالـ Router

R1# show arp					
Protocol	Address	Age (min)	Hardware Addr	Type	Interface
Internet	192.168.1.1	-	c202.5ea0.0000	ARPA	FastEthernet0/0
Internet	192.168.1.3	37	c203.0e00.0000	ARPA	FastEthernet0/0

دلوقي ممكن احجز IP لـ R2 مثلًا عن طريق الاوامر التالية

```
R1(config)# ip dhcp pool Client1
R1(dhcp-config)# host 192.168.1.50 255.255.255.0
R1(dhcp-config)# client-identifier 01C203.0e00.0000
R1(dhcp-config)# default-router 192.168.1.1
R1(dhcp-config)# dns-server 8.8.8.8
R1(dhcp-config)# exit
```

- اول حاجة بعمل DHCP Pool باسم Client1 مخصص للجهاز اللي هنثبتله IP.
- تحديد الـ IP الثابت اللي الجهاز هياخده (192.168.1.50).
- تحديد الـ MAC Address للجهاز (01C203.0e00.0000)، ولاحظ إننا بنضيف 01 في الأول لأنه بيشير لкарت Cisco Ethernet.
- إعداد الـ Options الخاصة بيه

لعمل Renew لفترة الـ Lease Manual على اجهزة Windows .. اكتب الامر التالي في الـ CMD

```
> ipconfig /renew
```

لحفظ الـ IP اللي خدته عن طريق الـ DHCP

```
> ipconfig /release
```

بعد تطبيق الامر دا، الجهاز مش هيبقى عنده عنوان IP وهينتقل للـ APIPA او الـ Alternate الا لو عملت اعادة تشغيل لкарتن الشبكة او كتبت الامر ipconfig /renew.

لحفظ الـ IP لكارتن شبكة معين

```
> ipconfig /release "Wi-Fi"
```

عرض كل الـ Pools الموجودة على الرواتر

R1#
R1#
R1# show ip dhcp pool

Pool LAN_POOL :		
utilization mark (high/low)	:	100 / 0
Subnet size (first/next)	:	0 / 0
Total addresses	:	254
Leased addresses	:	0
Pending event	:	none
1 subnet is currently in the pool :		
Current index	IP address range	Leased addresses
192.168.1.4	192.168.1.1 - 192.168.1.254	0

Pool Client1 :		
utilization mark (high/low)	:	100 / 0
Subnet size (first/next)	:	0 / 0
Total addresses	:	1
Leased addresses	:	1
Pending event	:	none
0 subnet is currently in the pool :		
Current index	IP address range	Leased addresses
192.168.1.50	192.168.1.50 - 192.168.1.50	1

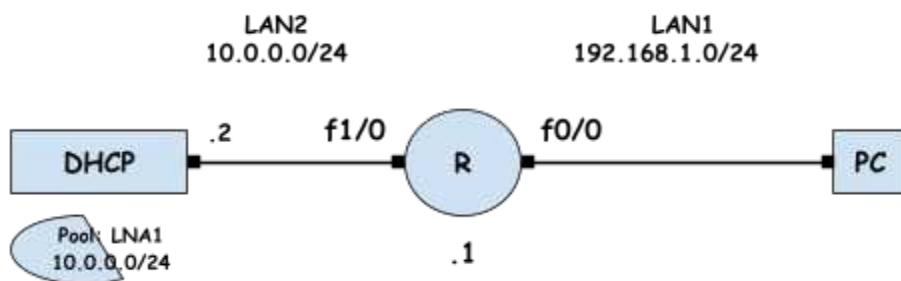
R1#

عرض عملية الـ DORA من خلال الـ Debugging

```
R1# debug ip dhcp server packet
```

18.2. IP-Helper Address

يستخدم علشان يوجه رسائل DHCP Broadcasts من شبكة معينة لشبكة تانية، وده بيسمح لأجهزة في شبكة فرعية إنها تأخذ IP من DHCP Server موجود في شبكة مختلفة.



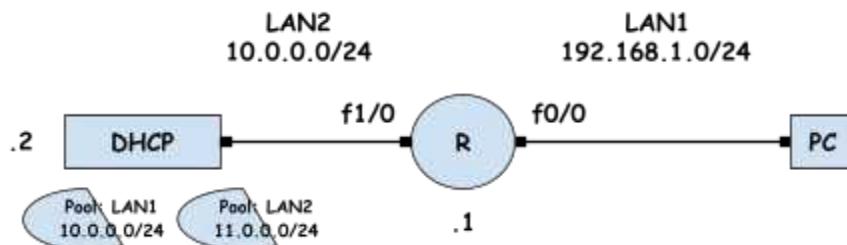
عشان الـ PC يأخذ IP يبعث أول DHCP Discover packet اللي هي Dynamic **Discover** بطريقة **Broadcast**. والراوتر مش بيعدى الـ Future Broadcast على IP-helper اسمها **IP-helper Address** .
الراوتر

- لما اطبق الـ Future دي .. أي Broadcast Packet يرسلها **raouter** للـ IP Unicast هيبعتها **Client** للـ IP الذي يطلب
- تم تحديده ك IP-helper Address
- وبالتالي هنحدد الـ IP الخاص بال DHCP على الـ Router IP-Helper Address . وبكدا الـ Client يحصل للـ IP الذي يطلب
- لتحديد IP-Helper Address

```
R1(config)# interface f0/0
R1(config-if)# ip address 192.168.1.1 255.255.255.0
R1(config-if)# ip helper-address 10.0.0.2
R1(config-if)# no shutdown
```

طيب لو عندي اكتر من 1 DHCP Server .. بيعرف ازاي ايه ال Pool المناسب للرد على طلب

ال Client



عشان ال DHCP يبعث ال IP المناسب لل Client <> الرواتر بيضيف ال IP الخاص بيه من ناحية ال

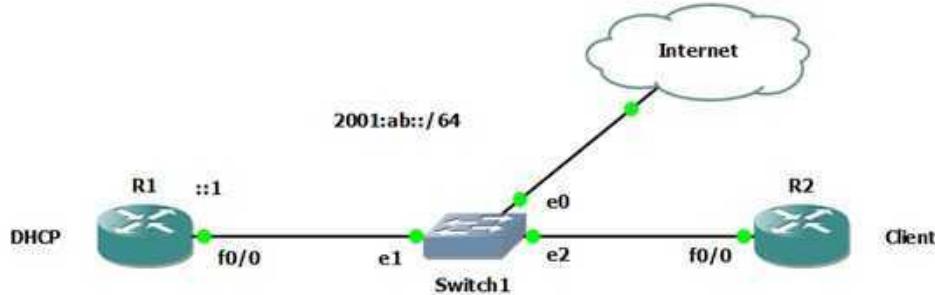
DHCP Packet في خانة اسمها GIADDR بتعال ال Header Client

كل Option بيتم توزيعه من خلال ال DHCP بيكون له رقم .. وال GIADDR بيأخذ رقم 82

لو عندك أكثر من DHCP Server، تقدر تضيف أكثر من IP Helper-Address، وهتشتغل Load Balancing بين السيرفرات.

```
R1(config)# interface f0/0
R1(config-if)# ip address 192.168.1.1 255.255.255.0
R1(config-if)# ip helper-address 10.0.0.2
R1(config-if)# ip helper-address 192.168.2.20
R1(config-if)# no shutdown
```

19. DHCPv6



19.1. Static IPv6 Configuration

لتفعيل IPv6 على الـ Interface Generate عشان يعمل IPv6 على الـ Link Local

```
R1(config)# interface f0/0
R1(config-if)# ipv6 enable
R1(config-if)# no shutdown
```

للتحقق

```
R1# show ipv6 interface brief
FastEthernet0/0           [up/up]
    FE80::C002:5EFF:FEA0:0
FastEthernet0/1           [administratively down/down]
```

لإعداد Manual Link Local IPv6 بشكل

```
R1(config)# interface f0/0
R1(config-if)# ipv6 address FE80::1 link-local
R1(config-if)# no shutdown
```

لإعداد Static IPv6

```
R1(config)# interface f0/0
R1(config-if)# ipv6 address 2001:db8:1::1/64
R1(config-if)# no shutdown
```

ملحوظة: لو اضفت Static IPv6 بشكل Manual .. مش لازم تكتب امر Enable ipv6 وهو تلقائياً هي فعل Manual IP وتحفظ بيه بجانب الـ Link Local IPv6 لـ Generate IPv6 وهي تعمل

للتحقق

```
R1# show ipv6 interface b
FastEthernet0/0          [up/up]
  FE80::C002:5EFF:FEA0:0
  2001:DB8:1::1
FastEthernet0/1          [administratively down/down]
```

19.2. Enable IPv6 Routing

لبدء استخدام الـ IPv6 Routing، أول حاجة لازم نفعل Dynamic Methods

عملية الـ Routing في راوترات Cisco لـ IPv4 تكون مفعولة تلقائياً، لكن لـ IPv6 لازم تستخدم الأمر

```
R1(config)# ipv6 unicast-routing
```

الامر دا هيخلني الـ Router يفعل خاصية التوجيه لـ IPv6 وبالتالي يقدر يوجه IPv6 Packets بين الـ Interfaces والشبكات المختلفة ويدعم الـ Dynamic Routing Protocols .. وكمان عشان الراوتر يقدر بيعدtPrefixes اللي من خالها بيوزع الـ Router Advertisement messages

19.3. Configure SLAAC

الطريقة دي بتخليلي أجهزة الـ Clients تعمل Dynamic GUA بشكل Generate بدون الحاجة لـ DHCP، أما الـ Client Server او الـ Prefixes هياخدوا من الرووتر اللي بيوزع Prefix

إعدادات R1 على SLAAC

- اول حاجة تفعيل IPv6 Routing

```
R1(config)# ipv6 unicast-routing
```

- إعدادات الـ Interface اللي هيوزع Prefix

```
R1(config)# int f0/0
R1(config-if)# ipv6 address FE80::1/64 link-local
R1(config-if)# ipv6 address 2001:db8:1::1/64
R1(config-if)# no shutdown
```

بعد تفعيل الـ Routing وب مجرد تحديد GUA على الـ Interface، الرووتر هيفعل الـ Router Advertisement Message .. وبالتالي الاجهزه هتقدر تاخذ الـ Prefix من الـ Interface دا

- للتحقق

```
R1(config-if)#do show ipv6 interface f0/0
FastEthernet0/0 is up, line protocol is up
IPv6 is enabled, link-local address is FE80::1
No virtual link-local address(es):
Global unicast address(es):
  2001:1234:A:B::1, subnet is 2001:1234:A:B::/64
Joined group address(es):
  FF02::1
  FF02::2
  FF02::1:FF00:1
MTU is 1500 bytes
ICMP error messages limited to one every 100 milliseconds
ICMP redirects are enabled
ICMP unreachables are sent
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds (using 30000)
ND advertised reachable time is 0 (unspecified)
ND advertised retransmit interval is 0 (unspecified)
ND router advertisements are sent every 200 seconds
ND router advertisements live for 1800 seconds
ND advertised default router preference is Medium
Hosts use stateless autoconfig for addresses.
```

```
R2(config)# interface f0/0
R2(config-if)# ipv6 address autoconfig default
R2(config-if)# no shutdown
```

```
R1 | R2 x
Router(config-if)#
Router(config-if)#do show ipv6 interface f0/0
FastEthernet0/0 is up, line protocol is up
  IPV6 is enabled, link-local address is FE80::C003:EFF:FE00:0
  No virtual link-local address(es):
  Global unicast address(es):
    2001:1234:A:B::C003:EFF:FE00:0, subnet is 2001:1234:A:B::/64 [EUI/CAL/PRE]
      valid lifetime 2591891 preferred lifetime 604691
  Joined group address(es):
    FF02::1
    FF02::1:FF00:0
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ICMP unreachables are sent
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  Default router is FE80::1 on FastEthernet0/0
Router(config-if)#

```

19.4. Configure Stateless DHCPv6 with SLAAC

هنتطبق نفس اعدادات الـ SLAAC عشان الـ Clients يقدرو يحصلو على Prefix من الراوتر .. وكمان هنعمل .Domain Name Options زي الـ DNS والـ Options لـ DHCPv6 لـ Configuration

إعدادات الـ Server

- اول حاجة تفعيل IPv6 Routing

```
R1(config)# ipv6 unicast-routing
```

- إعدادات DHCPv6

إنشاء الـ DHCPv6 Pool مع تحديد الـ Options

```
R1(config)# ipv6 dhcp pool LAN
R1(config-dhcpv6)# dns-server 2001:ab::5
R1(config-dhcpv6)# domain-name tshoot.com
```

ربط الـ Interface بالـ Pool

```
R1(config)# int f0/0
R1(config-if)# ipv6 address FE80::1/64 link-local
R1(config-if)# ipv6 address 2001:db8:1::1/64
R1(config-if)# ipv6 nd other-config-flag
R1(config-if)# ipv6 dhcp server LAN
R1(config-if)# no shutdown
```

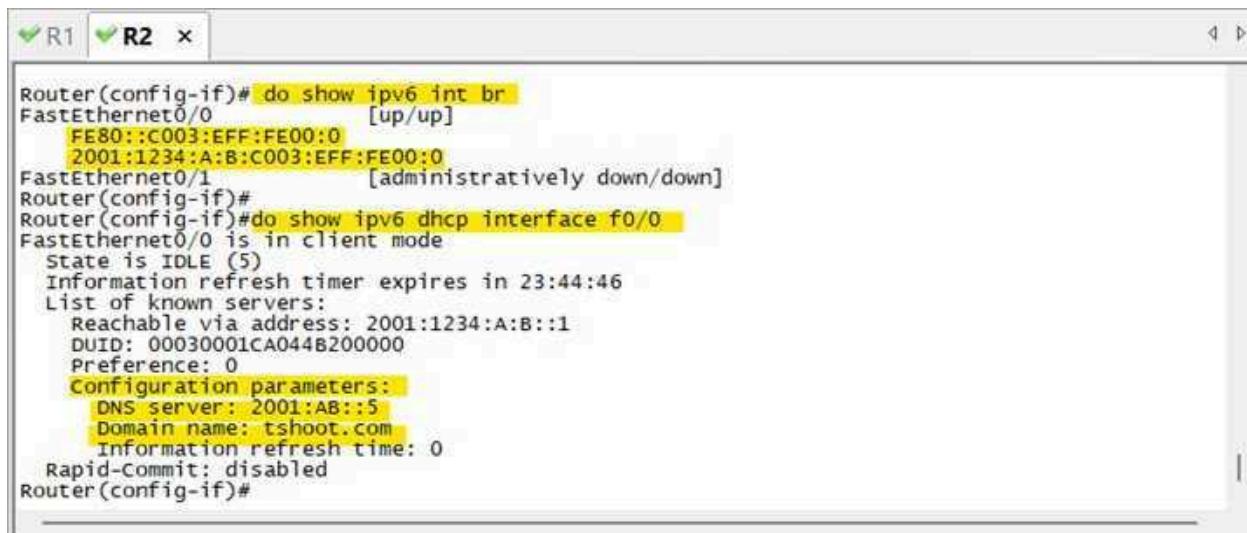
أمر `ipv6 nd other-config-flag` بيكقول للأجهزة إنها تأخذ الـ IP من SLAAC، لكن

.DHCPv6 زي الـ DNS من معلومات إضافية زي الـ Options

اعدادات الـ Client

```
R2(config)# interface f0/0
R2(config-if)# ipv6 address autoconfig default
R2(config-if)# no shutdown
```

للحقيق



The screenshot shows a Cisco Packet Tracer interface with two routers, R1 and R2. Router R2 is selected. The configuration window displays the following commands:

```
R2(config-if)# do show ipv6 int br
FastEthernet0/0 [up/up]
    FE80::C003:EFF:FE00:0
    2001:1234:A:B:C003:EFF:FE00:0
FastEthernet0/1 [administratively down/down]
Router(config-if)#
Router(config-if)#do show ipv6 dhcp interface f0/0
FastEthernet0/0 is in client mode
State is IDLE (5)
Information refresh timer expires in 23:44:46
List of known servers:
    Reachable via address: 2001:1234:A:B::1
    DUID: 00030001CA044B200000
    Preference: 0
    Configuration parameters:
        DNS server: 2001:A8::5
        Domain name: tshoot.com
        Information refresh time: 0
    Rapid-Commit: disabled
Router(config-if)#

```

19.5. Configure Stateful DHCPv6

إعدادات الـ **DHCPv6 Server** لتوزيع الـ IP والـ Prefix وباقی الـ Options

إعدادات الـ **Server**

- اول حاجة تفعيل IPv6 Routing

```
R1(config)# ipv6 unicast-routing
```

- إعدادات DHCPv6

إنشاء الـ Pool مع تحديد الـ Options

```
R1(config)# ipv6 dhcp pool LAN
R1(config-dhcp)# address prefix 2001:db8:1::0/64
R1(config-dhcp)# dns-server 2001:db8:1::5
R1(config-dhcp)# domain-name tshoot.com
```

ربط الـ Pool بالـ Interface

```
R1(config)# int f0/0
R1(config-if)# ipv6 address 2001:db8:1::1/64
R1(config-if)# ipv6 dhcp server LAN
R1(config-if)# ipv6 nd managed-config-flag
R1(config-if)# ipv6 nd prefix 2001:db8:1::/64 no-autoconfig
R1(config-if)# no shutdown
```

أمر **ipv6 nd prefix 2001:db8:1::/64** و **ipv6 nd managed-config-flag** .DHCPv6 Clients يخلو الـ **no-autoconfig** ويأخذون العنوان من SLAAC ميستخدموش

```
R2(config)# interface f0/0
R2(config-if)# ipv6 address dhcp
R2(config-if)# no shutdown
```

جربت طرق كتير مع استخدام الأمر `ipv6 address autoconfig default` عشان الرووتر ياخذ IP من الـ DHCP و ميستخدمش SLAAC، ولكن فالنهاية كان بعتمد على SLAAC برضو .. وفي بعض المصادر كان بيستخدم امر `ipv6 address dhcp`

لما جربت `ipv6 address dhcp` على راوترات:

- 7200 Software (C7200-ADVIPSERVICESK9-M), Version 15.2(4)S5
- Software, 3700 Software (C3725-ADVENTERPRISEK9-M), Version 12.4(15)T5

لقيت الامر مش مدعاوم .. فجريت استخدم الامر على

I86BI_LINUX-ADVENTERPRISEK9-M), Version 15.7(3)M2)

وكان مدعاوم وخد IP من الـ DHCP

<https://www.networkacademy.io/ccna/ipv6/stateful-dhcpv6>

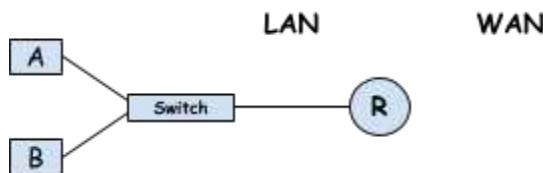
<https://forum.networklessons.com/t/cisco-dhcpv6-server-configuration/1116/19?page=2>

20. Access Control List

الـ ACL هي مجموعة من القواعد Rules او الجمل البرمجية اللي بتتطبق على الـ Interfaces في الراوتر أو السويتش، وبتحدد الـ Traffic اللي مسموح ليه يعدي والـ Traffic اللي هيتمنعه. القواعد دي بتشتغل بالترتيب، وده بيساعد في تحسين الأمان وتقليل استهلاك الموارد.

أمثلة:

- ممكن احدد ان جهاز A يقدر يصل للـ WAN، وجهاز B غير مسموح
- ممكن اسمح بمواقع معينة واعمل حظر لمواقع معينة
- ممكن اسمح لجهاز A بالوصول للـ Router من خلال SSH فقط .. والسماح لجهاز B بالوصول لـ Telnet فقط من خلال Router



أنواع الـ ACL

- بسيطة Standard ACL:
 - بتفلتر الـ Traffic بناء على الـ Source IP Address فقط. يعني لو الجهاز متصل من IP معين، يقدر يعدي أو لا او يقدر يصل للـ Internet ولا لا، لكن متقدرش تمنع او تسمح بموقع او بروتوكولات او Ports معينة.

Numbered ◦

Named ◦

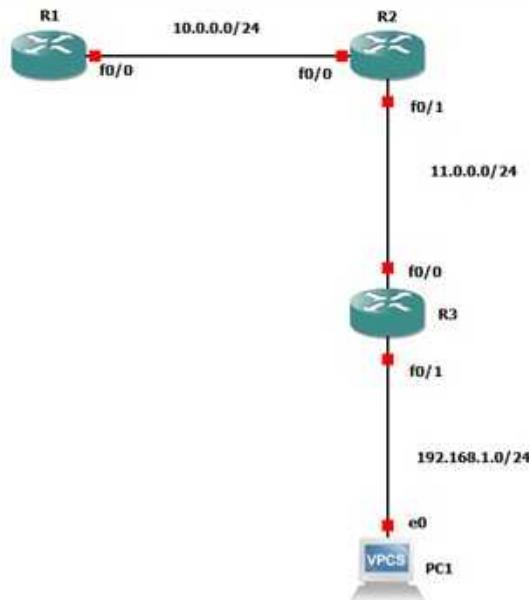
- متقدمة Extended ACL

بتفلتر الـ Traffic بناء على الـ Source & Destination IP وكمان البروتوكولات (TCP, UDP, ICMP, etc) والمنافذ (Ports).

Numbered ◦

Named ◦

20.1. Access-List LAB



20.1.1. Configure Routing

أول حاجة نقوم

R1

```
R1(config)# int f0/0
R1(config-if)# ip add 10.0.0.1 255.255.255.0
R1(config-if)# no sh
R1(config-if)# ip ospf 1 area 0
```

R2

```
R2(config)# int f0/0
R2(config-if)# ip add 10.0.0.2 255.255.255.0
R2(config-if)# no sh
R2(config-if)# ip ospf 1 area 0
R2(config)# int f0/1
R2(config-if)# ip add 11.0.0.2 255.255.255.0
R2(config-if)# no sh
R2(config-if)# ip ospf 1 area 0
```

R3

```
R3(config)# int f0/0
R3(config-if)# ip add 11.0.0.3 255.255.255.0
R3(config-if)# no sh
R3(config-if)# ip ospf 1 area 0
R3(config)# int f0/1
R3(config-if)# ip add 192.168.1.3 255.255.255.0
R3(config-if)# no sh
R3(config-if)# ip ospf 1 area 0
```

PC

```
PC1> ip 192.168.1.10 255.255.255.0 192.168.1.3
Checking for duplicate address...
PC1 : 192.168.1.10 255.255.255.0 gateway 192.168.1.3
```

Test Connectivity

✓ R1 | ✓ R2 | ✓ R3 | ✓ PC1 ×

```
PC1>
PC1>
PC1> ping 10.0.0.1
84 bytes from 10.0.0.1 icmp_seq=1 ttl=253 time=79.415 ms
84 bytes from 10.0.0.1 icmp_seq=2 ttl=253 time=80.423 ms
84 bytes from 10.0.0.1 icmp_seq=3 ttl=253 time=79.542 ms
84 bytes from 10.0.0.1 icmp_seq=4 ttl=253 time=79.138 ms
84 bytes from 10.0.0.1 icmp_seq=5 ttl=253 time=78.782 ms
PC1>
```

✓ R1 × | ✓ R2 | ✓ R3 | ✓ PC1

```
.!!!!!
Success rate is 80 percent (4/5), round-trip min/avg/max = 76
/336/1112 ms
R1#
R1#ping 192.168.1.10
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.10, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 72/79/88 ms
R1#
```

20.1.2. Standard ACL Configuration

مثلا لو عايزين نسمح بالـ Traffic اللي جاي من R2 على R1

اول حاجة بنكتب جملة الـ ACL في الـ Configuration Mode وبعدين بنطبقها تحت اي Interface

```
R1(config)# access-list 1 permit 10.0.0.0 0.0.0.255
```

شرح الأمر:

- اول حاجة اكتب كلمة access-list متبوعة برقم الـ ACL

نلاحظ ان الـ Standard ACL ممكن تاخد رقم من الـ Range دا

```
R1(config)#  
R1(config)# access-list ?  
<1-99>          IP standard access list  
<100-199>        IP extended access list  
<1100-1199>      Extended 48-bit MAC address access list  
<1300-1999>      IP standard access list (expanded range)  
<200-299>        Protocol type-code access list  
<2000-2699>      IP extended access list (expanded range)  
<2700-2799>      MPLS access list  
<700-799>         48-bit MAC address access list  
compiled           Enable IP access-list compilation  
dynamic-extended  Extend the dynamic ACL absolute timer  
rate-limit         Simple rate-limit specific access list
```

- حدد نوع الـ access-list .. هل هي Permit .. و في نوع ثاني اسمه Deny

حدد الـ Source Network اللي هتسمح بيها او هتمنعها

اكتب الـ Wild Card اللي يحدد الشبكة او الـ Host

مثلا الامر permit 10.0.0.0 0.0.0.255 يسمح بشبكة 10.0.0.0 كاملة

وممكن تكتب permit 10.0.0.2 0.0.0.0 لتحديد IP واحد فقط

ولو هتحدد IP واحد فقط ممكن تكتب كلمة host قبل الـ IP بدل استخدام الـ Wild Card

permit host 10.0.0.1

ممكن تكتب كلمة log في نهاية الامر عشان يعرض Syslog كل ما يصل Packet مسمومة او

محظورة على حسب انتا كاتب Permit او Deny .. والأمر دا بيعمل Load على الـ CPU

access-list 1 permit 10.0.0.0 0.0.0.255 log

لو عايز تمنع الـ Traffic اللي جاي من R3 على R1 <> ممكن تكتب جملة او Rule مختلفة في نفس رقم الـ access-list

```
R1(config)# access-list 1 deny 11.0.0.0 0.0.0.255
```

لتطبيق الـ access-list

```
R1(config)# interface f0/0
R1(config-if)# ip access-group 10 in
```

```
R1(config)# access-list 1 permit 10.0.0.0 0.0.0.255
R1(config)# access-list 1 deny 11.0.0.0 0.0.0.255
R1(config)#
R1(config)#int f0/0
R1(config-if)#ip access-group 1 in
R1(config-if)#ip access-group 1 ?
    in    inbound packets
    out   outbound packets
```

ايه الفرق بين in و out

مثلا الـ role اللي بتعملها للـ Traffic اللي جاي من R3

- ممكن تطبقها على Interface f0/0 وتكتب in <> يعني تمنع الـ Traffic من الدخول على .. وفي الحالة دي الرووتر اول ما يشوف الـ Source Network الموجودة في الـ Interface f0/0 هيعملها للـ access-list Packet Drop
- ممكن تطبقها على Interface f0/1 لو موجود وتكتب out <> يعني تمنع الـ Traffic من الخروج من .. وفي الحالة دي الرووتر هي تعالج الـ Packet الاول ويشوف هتخرج من اي Interface f0/1 وبعدين يعملها Drop لو هتخرج من Interface f0/1

Test

```
R1 R2 R3 PC1
R2#
R2#
*Mar 14 17:32:05.363: %SYS-5-CONFIG_I: Configured from console by console
R2#
R2# ping 10.0.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.0.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/32/40 ms
R2#
```

```
R1 R2 R3 PC1
R3#
R3#
*Mar 14 17:31:46.227: %SYS-5-CONFIG_I: Configured from console by console
R3#
R3# ping 10.0.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.0.1, timeout is 2 seconds:
UUUUU
Success rate is 0 percent (0/5)
R3#
```

حرف الـ U يعني Unreachable

حرف الـ U ممکن یعرف المستخدم ان في access-list Traffic مطبقة وبمتنع الـ الخاص بيه .. وبالتالي ممکن نستخدم الامر التالي بحيث المستخدم میعرفش ایه نوع المشكلة

```
R1(config)# int f0/0
R1(config-if)# no ip unreachable
```

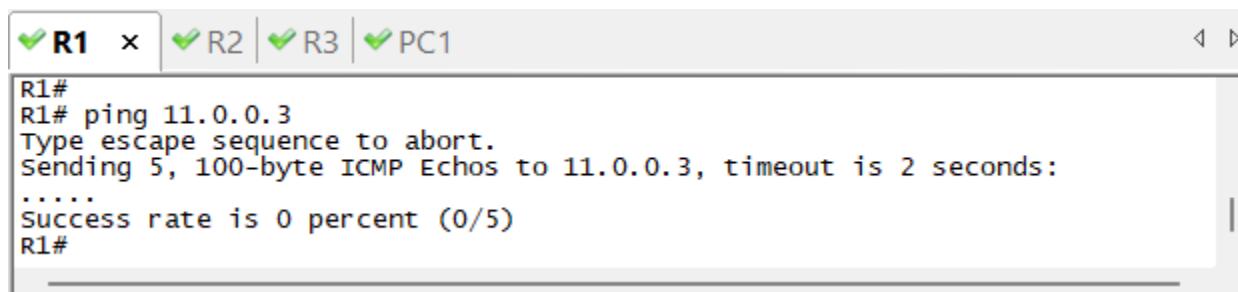
```
R1 R2 R3 PC1
R3#
R3# ping 10.0.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.0.1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
R3#
```

ملاحظة:

لو كتبت `log` في الـ `access-list rule` وفي `Packet` بتوصيل بشكل متكرر، زي الـ OSPF Hello Packet اللي بتوصيل كل 10 ثواني >> أول مره هتظهر وبعدين مش هتظهر تاني غير بعد 5 دقائق طالما جاية من نفس الـ Source وبنفس القيم ومتطابقة تماما مع اللي قبلها.

إبما اننا منعنا الـ Traffic اللي جاي من `R1` على `R3` وبالفعل الـ `R1` بتوصيل لـ `R1` لكنه بعملها `! Reply` ومش بيرد بـ `Reply` .. هل لو عملنا `Ping` من `R1` على `R3`، هيوصل `Drop`

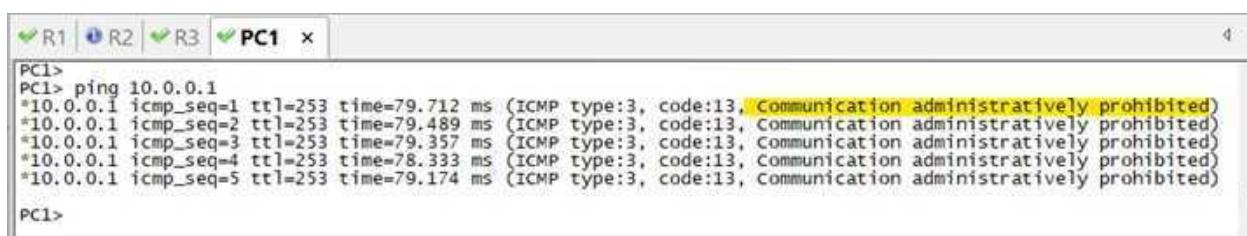
• الـ `Packet` هتوصل عادي من `R1` لـ `R3` .. لكن لما `R3` يرد بـ `Reply` .. الـ `Reply` هيوصل لـ `R1` لكن `Access-List` بسبب الـ `Drop` بعمله



```
R1# ping 11.0.0.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.0.0.3, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
R1#
```

!! طيب لو عملنا `Ping` من `R1` على `PC1`، هيوصل `! Reply` !!

• دايما خر `Rule` في الـ `ACL` بتكون "**deny any**" النظم بيضيفها بشكل تلقائي، يعني أي `Traffic` مش مطابق للـ `Rules` اللي حدتها بيعمله `Drop`.



```
PC1> ping 10.0.0.1
*10.0.0.1 icmp_seq=1 ttl=253 time=79.712 ms (ICMP type:3, code:13, Communication administratively prohibited)
*10.0.0.1 icmp_seq=2 ttl=253 time=79.489 ms (ICMP type:3, code:13, Communication administratively prohibited)
*10.0.0.1 icmp_seq=3 ttl=253 time=79.357 ms (ICMP type:3, code:13, Communication administratively prohibited)
*10.0.0.1 icmp_seq=4 ttl=253 time=78.333 ms (ICMP type:3, code:13, Communication administratively prohibited)
*10.0.0.1 icmp_seq=5 ttl=253 time=79.174 ms (ICMP type:3, code:13, Communication administratively prohibited)
PC1>
```

لاحظ انه مكتوب الـ `Communication Pobibted` يعني ممنوع، والمفروض ميكونش في `Reply` ودا اللي بيحصل في الـ `Production`

عرض الـ access-list

```
R1# show access-list
Standard IP access list 1
    10 deny   11.0.0.0, wildcard bits 0.0.0.255 (10 matches)
    20 permit 10.0.0.0, wildcard bits 0.0.0.255 (333 matches)
R1#
```

- هنلاقي الـ Rules اللي كتبناها في الـ access-list اللي رقمها 1
- كل Rule لها رقم اسمه Sequence Number وكل سطر بنسمهه Access Control Entry
- اي Traffic يوصل للـ Router يقارنو بالـ Rules المكتوبة بالترتيب، ويطبق عليه او Rule يحصلها Match
 - يعني مثلا لو Traffic جاي من شبكة 192.168.1/24
 - هيقارنو باول Rule ..» مش هيحصل Match
 - هيقارنو بتاني Rule ..» مش هيحصل Match
 - لو محصل Match مع اي Rule هيعمل Drop لـ Packet لأن آخر Rule دايما بتبقى Deny
 - اللي بتمنع اي Traffic من اي شبكة Any
- ممكن اخلي اخر Rule تسمح باي Traffic عن طريق كتابة access-list 1 permit any
- ترتيب كتابة الـ Rule مهم جدا
 - مثلا لو كتبت 11.0.0.0/24 access-list 1 deny 11.0.0.0 0.0..0.255 لمنع شبكة 11.0.0.1
 - وبعدين كتبت access-list 1 Permit host 11.0.0.3 للسماح بالـ IP دا فقط
 - اللي هيحصل ان لو Traffic جاي من 11.0.0.1 هيقارن باول Rule وبالتالي هيتعمل Drop
 - مباشرة .. لأن بمجرد ما يحصل Match مع اي Rule .. الـ Rule مش بيبص على اي تانية
 - وبالتالي بنكتب الـ Rule الاكثر تحديدا الاول

Remark Action

الـ **Remark** في الـ **ACL** بيستخدم عشان تضيف تعليقات توضيحية جوه الـ **ACL**, بحيث تبقى القواعد واضحة وسهلة الفهم لما ترجع لها بعد فترة.

```
R1(config)# access-list 1 remark this ACL was to be applied to int f0/0
```

The screenshot shows the Cisco IOS command-line interface. At the top, there are tabs for R1, R2, R3, and PC1, with R1 selected. The main window displays the following configuration:

```
R1(config)#  
R1(config)#access-list 1 remark this ACL was to be applied to int f0/0  
R1(config)#  
R1(config)# do show run | in access-list  
access-list 1 deny 11.0.0.0 0.0.0.255  
access-list 1 permit 10.0.0.0 0.0.0.255  
access-list 1 remark this ACL was to be applied to int f0/0  
R1(config)#
```

مشكلة الـ Numbered Access-List

- لما تلغي Rule معينة <> دا هيادي الى مسح كل الـ Rules الموجودة فالـ **Access-List** .. عشان كدا لازم تأخذ بالك من الغاء الـ **Access-List** تحت الـ **Interface** لو مبقتش مستخدمة، عشان لو اضفت فيها Rules في المستقبل متطبقش مباشرة وانتا ناسي
- ومع ذلك الامر هيفضل مطبق تحت الـ **Interface** .. عشان كدا لازم تأخذ بالك من الغاء الـ **Access-List** تحت الـ **Interface** لو مبقتش مستخدمة، عشان لو اضفت فيها Rules في المستقبل

```

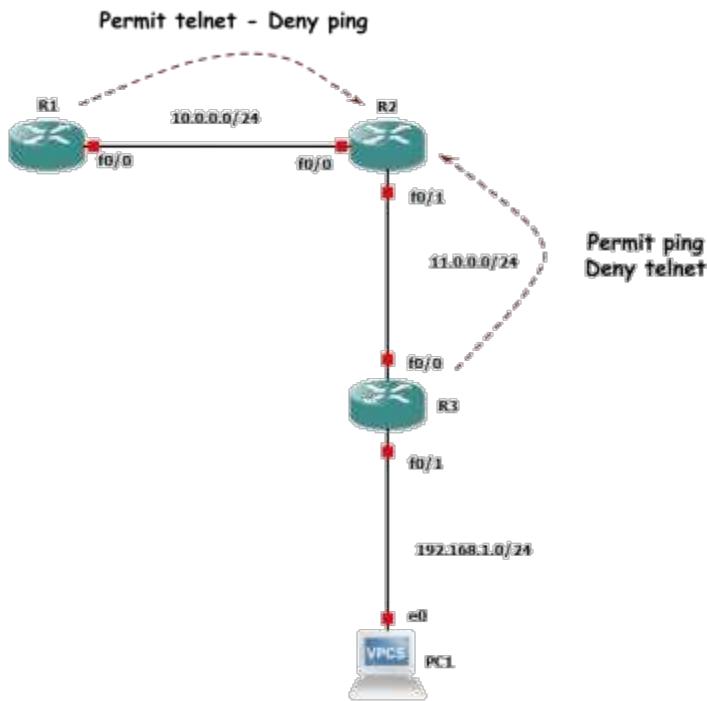
✓ R1 ✘ | ✓ R2 | ✓ R3 | ✓ PC1

R1(config)# do show run | in access-list
access-list 1 deny 11.0.0.0 0.0.0.255
access-list 1 permit 10.0.0.0 0.0.0.255
access-list 1 remark this ACL was to be applied to int f0/0
R1(config)#
R1(config)# no access-list 1 permit 10.0.0.0 0.0.0.255
R1(config)#
R1(config)# do show run | in access-list
R1(config)#
R1(config)# do show run int f0/0
Building configuration...

Current configuration : 121 bytes
!
interface FastEthernet0/0
 ip address 10.0.0.1 255.255.255.0
 ip access-group 1 in
 ip ospf 1 area 0
 duplex full

```

20.1.3. Extended ACL Configuration



الهدف

- السماح لـ R1 بالوصول لـ R2 عن طريق بروتوكول telnet، ومنعه من الوصول عن طريق الـ Ping
- السماح لـ R3 بالوصول لـ R2 عن طريق بروتوكول Ping، ومنعه من الوصول عن طريق Telnet

الـ Extended ACL المحدد للـ Range

```
R1 | R2 | R3 | PC1
R2(config)# access-list ?
<1-99>          IP standard access list
<100-199>        IP extended access list
<1100-1199>      Extended 48-bit MAC address access list
<1300-1999>      IP standard access list (expanded range)
<200-299>         Protocol type-code access list
<2000-2699>       IP extended access list (expanded range)
<2700-2799>       MPLS access list
<700-799>          48-bit MAC address access list
compiled          Enable IP access-list compilation
dynamic-extended  Extend the dynamic ACL absolute timer
rate-limit        Simple rate-limit specific access list
R2(config)# access-list
```

كدا هنسخدم access-list نوعها عشان نقدر نحدد Source و Destination وببروتوكول .. وال Ping دى هيكون فيها ثلاثة Rules بالنسبة لـ R1؛ واحدة للسماح بال Telnet والثانية لمنع الـ Access-List والثالثة للسماح ببروتوكول OSPF عشان الشبكة تشغل كوييس

```
R2(config)# access-list 100 permit tcp host 10.0.0.1 host 10.0.0.2 eq telnet
R2(config)# access-list 100 permit icmp host 10.0.0.1 host 10.0.0.2 echo
R2(config)# access-list 100 permit ospf any any
```

شرح اول Rule

- اكتب الكلمة Extended Range وحدد رقمها من الـ Access-list
- حدد الـ Action اللي هيطبق <> Permit
- حدد الـ Protocol Group بالاسم او الرقم .. يعني اي بروتوكول بيقى مختلف في TCP او UDP او هو ICMP منفصل زي الـ OSPF و الـ

```
R2(config)# access-list 100 permit ?
<0-255> An IP protocol number
ahp Authentication Header Protocol
eigrp Cisco's EIGRP routing protocol
esp Encapsulation Security Payload
gre Cisco's GRE tunneling
icmp Internet Control Message Protocol
igmp Internet Gateway Message Protocol
ip Any Internet Protocol
ipinip IP in IP tunneling
nos KA9Q NOS compatible IP over IP tunneling
ospf OSPF routing protocol
pcp Payload Compression Protocol
pim Protocol Independent Multicast
tcp Transmission Control Protocol
udp User Datagram Protocol
```

- حدد الـ Source Host او الـ Source Network
- حدد الـ Destination Network/Host
- حدد البروتوكول او الـ Port Number لو اسم البروتوكول مش موجود في الـ Help
 - ممكن تستخدم eq لتحديد بروتوكول واحد فقط
 - ممكن تستخدم gt → greater than لتحديد كل البروتوكولات اللي اكبر من رقم معين او le لتحديد كل البروتوكولات اللي اصغر من رقم معين
 - وفي اختيارات تانية كتير

```
R2(config)# access-list 100 permit tcp host 10.0.0.1 host 10.0.0.2 ?
ack Match on the ACK bit
dscp Match packets with given dscp value
eq Match only packets on a given port number
established Match established connections
fin Match on the FIN bit
fragments Check non-initial fragments
gt Match only packets with a greater port number
log Log matches against this entry
log-input Log matches against this entry, including input interface
lt Match only packets with a lower port number
neq Match only packets not on a given port number
precedence Match packets with given precedence value
psh Match on the PSH bit
range Match only packets in the range of port numbers
rst Match on the RST bit
syn Match on the SYN bit
time-range Specify a time-range
tos Match packets with given TOS value
urg Match on the URG bit
<CR>
```

بالنسبة لـ Role الثانية

```
R2(config)# access-list 100 permit icmp host 10.0.0.1 host 10.0.0.2 echo
```

استخدم echo reply لمنع الـ echo request والـ

لتطبيق الـ Access-List

```
R2(config)# int f0/0
R2(config-if)# ip access-list 100 in
```

الـ R3 الخاصة بـ Access-List

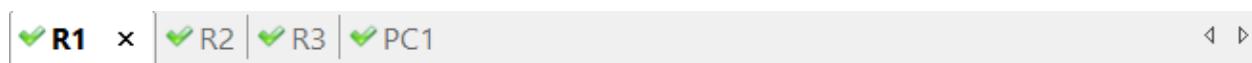
```
R2(config)# access-list 101 deny tcp host 11.0.0.3 host 11.0.0.2 eq telnet
R2(config)# access-list 101 permit icmp host 11.0.0.3 host 11.0.0.2 echo
R2(config)# access-list 101 permit ospf any any
R2(config)# interface f0/1
R2(config-if)# ip access-group 101 in
```

للسماح بالـ Telnet على R2

```
R2(config)# line vty 0 4
R2(config)# no login
```

الامر دا يجب تجنبه تماما في البيئة العملية لانه مش يطلب User ولا Pass للدخول على الراوتر من الـ Telnet

Test



```
R1# ping 10.0.0.2
R1# Type escape sequence to abort.
      Sending 5, 100-byte ICMP Echos to 10.0.0.2, timeout is 2 seconds:
      UUUUU
      Success rate is 0 percent (0/5)
R1#
R1# telnet 10.0.0.2
Trying 10.0.0.2 ... Open
R2>
R2>
```

```

R3# ping 11.0.0.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.0.0.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 24/37/64 ms
R3# telnet 11.0.0.2
Trying 11.0.0.2 ...
% Destination unreachable; gateway or host down
R3#

```

20.1.4. Named ACL

مشكلة الـ ACL Numbered بشكل عام، سواء Standard او Extended ان لو لغت Rule او Rule بحذف الـ Named ACL يعني دائما بتتضاف في الآخر. اما الـ Access List بتدعم الحذف وبتدعم الـ Insert .. يعني اضافة Rule في مكان معين

لكتابة Named ACL

```
R2(config)# ip access-list standard CISCO
R2(config-std-nacl)#

```

• اكتب ip access-list متبوعة بنوع الـ Access-list

```
R2(config)# ip access-list ?
extended          Extended Access List
helper            Access List acts on helper-address
log-update        Control access list log updates
logging           Control access list logging
match-local-traffic Enable ACL matching for locally generated traffic
resequence         Resequence Access List
role-based        Role-based Access List
standard          Standard Access List
```

• ثم حدد رقم او اسم للـ ACL، الاسم بيكون Case Sensitive

```
R2(config)# ip access-list st
R2(config)# ip access-list standard ?
<1-99>      Standard IP access-list number
<1300-1999>  Standard IP access-list number (expanded range)
WORD          Access-list name
```

- وبعدين اضغط Enter واتكتب الـ Rules اللي انتا عايزةها

```
R2(config)# ip access-list standard CISCO
R2(config-std-nacl)# permit 10.0.0.0 0.0.0.255
R2(config-std-nacl)# deny 11.0.0.0 0.0.0.255
R2(config-std-nacl)# permit host 1.1.1.1
R2(config-std-nacl)# deny host 2.2.2.2
```

عرض الـ ACLs المتاحة

```
R2# show access-list
Standard IP access list CISCO
 30 permit 1.1.1.1
 40 deny 2.2.2.2
 10 permit 10.0.0.0, wildcard bits 0.0.0.255
 20 deny 11.0.0.0, wildcard bits 0.0.0.255
Extended IP access list 100
 10 permit tcp host 10.0.0.1 host 10.0.0.2 eq telnet (39 matches)
 20 deny icmp host 10.0.0.1 host 10.0.0.2 echo (5 matches)
 30 permit ospf any any (249 matches)
Extended IP access list 101
 10 deny tcp host 11.0.0.3 host 11.0.0.2 eq telnet (1 match)
 20 permit icmp host 11.0.0.3 host 11.0.0.2 echo (15 matches)
 30 permit ospf any any (185 matches)
```

- أمر `show access-list` هيعرض كل الـ ACLs اللي كتبتها
- نلاحظ في الـ ACL انها مش معروضة بالترتيب، ودا لانه بيعرض الـ Rules الاكثر تحديدا
- الاول، بس التنفيذ بيتم على حسب الـ Sequence Number
- نلاحظ ان بين كل Rule عشر ارقام، ودا عشان نقدر نعمل Rule Insert لـ Rule بينهم .. وكمان نقدر نغير الفرق دا ونقدر نعيد ترتيبهم وكان نقدر نستخدم Number ACL ونعدل عليها

للغاء Rule محددة

```
R1 R2 R3 PC1
R2(config)# do show access-list
Standard IP access list CISCO
 30 permit 1.1.1.1
 40 deny  2.2.2.2
 10 permit 10.0.0.0, wildcard bits 0.0.0.255
 20 deny   11.0.0.0, wildcard bits 0.0.0.255
Extended IP access list 100
 10 permit tcp host 10.0.0.1 host 10.0.0.2 eq telnet (39 matches)
 20 deny  icmp host 10.0.0.1 host 10.0.0.2 echo (5 matches)
 30 permit ospf any any (310 matches)
Extended IP access list 101
 10 deny tcp host 11.0.0.3 host 11.0.0.2 eq telnet (1 match)
 20 permit icmp host 11.0.0.3 host 11.0.0.2 echo (15 matches)
 30 permit ospf any any (246 matches)
R2(config)#
R2(config)# ip access-list standard CISCO
R2(config-std-nacl)# no 10
R2(config-std-nacl)#
R2(config-std-nacl)# ip access-list standard CISCO
R2(config-std-nacl)# do show access-list
Standard IP access list CISCO
 30 permit 1.1.1.1
 40 deny  2.2.2.2
 20 deny   11.0.0.0, wildcard bits 0.0.0.255
```

لإضافة Rule برقم معين Sequence Number

```
R2(config)# ip access-list standard CISCO
R2(config-std-nacl)# 25 permit host 10.0.0.1
R2(config)# do show access-list
Standard IP access list CISCO
 30 permit 1.1.1.1
 40 deny  2.2.2.2
 25 permit 10.0.0.1
 20 deny   11.0.0.0, wildcard bits 0.0.0.255
```

لزيادة الفارق بين كل Rule

```
R2(config)# ip access-list resequence CISCO 10 20
R2(config)# do show access-list
Standard IP access list CISCO
 10 permit 1.1.1.1
 30 deny  2.2.2.2
 50 permit 10.0.0.1
 70 deny   11.0.0.0, wildcard bits 0.0.0.255
```

حيث 10 هي اول Rule تبدأ تزود من عندها .. و 20 هو الفارق الجديد بين كل Rule والثانية

للتعديل على Named ACL بطريقة إلـا

```
R2(config)# do show access-list
Extended IP access list 100
    10 permit tcp host 10.0.0.1 host 10.0.0.2 eq telnet (39 matches)
    20 deny icmp host 10.0.0.1 host 10.0.0.2 echo (5 matches)
    30 permit ospf any any (361 matches)
R2(config)# ip access-list extended 100
R2(config-ext-nacl)# no 20
R2(config-ext-nacl)# exit
R2(config)# do show access-list
Extended IP access list 100
    10 permit tcp host 10.0.0.1 host 10.0.0.2 eq telnet (39 matches)
    30 permit ospf any any (391 matches)
```

امتنى يفضل استخدام أمر **log**

يفضل تضييف في اخر إلـا ACL مع امر **log** اللي يكون موجود By Default اصلا بس مع اضافة **Log** عشان يكون عندي معلومة لو في اي Traffic تم منعه بسبب الامر دا.

```
R2(config)# ip access-list 100 deny ip any any log
```

20.1.5. Access-Class

إلـا Access-Class بيستخدم عشان يتحكم في الأجهزة اللي تقدر توصل للراوتر عن طريق Telnet أو SSH وبالتالي بدل ما بيتطبق على إلـا Interfaces، بيتطبق على إلـا VTY Lines عشان يحدد مين مسموح له يدخل على الراوتر عن بعد. يعني إلـا Access Class بيتحكم في إلـا Traffic اللي داخل لـا Router نفسه، مش اللي بيعدي من خلاله زي ما إلـا ACL العاديـة بتعمل.

لأن تطبيق إلـا Access-List العاديـة يسبب بعض المشاكل في إلـا Telnet والا SSH .. مثلا لو عايز اسمح بالTelnet من R1 لـ R2 وعايز امنع إلـا Telnet من R2 على R1 : في الحالة دي، إلـا Traffic العكسي اللي راجع من R2 على R1 هيبقى ممنوع

مثلا للسماح له R1 بالـ telnet على R2

```
R2(config)# access-list 1 permit host 10.0.0.1
R2(config-line)# line vty 0 4
R2(config-line)# access-class 1 in
```

لو عايز تمنع اللي فاتحين R2 من خلال الـ Console من الوصول له R1 .. مع السماح لهم بالوصول له R3

```
R2(config)# access-list 2 deny host 10.0.0.1
R2(config)# access-list 2 permit host 11.0.0.3
R2(config-line)# line console 0
R2(config-line)# access-class 2 out
```

20.1.6. Filter debugging using ACL

في بعض الأحيان، لما بنستخدم أوامر الـ Debugging، بتطلع كمية ضخمة جدا من الـ Syslogs، وده ممكن يكون صعب في التحليل أو حتى يؤثر على أداء الراوتر. مثلا استخدام امر debug ip packet detail لـ Overwhelm والـ Prompt Router داخلة على الراوتر وممكن يعمل له Packet Overwhelm بحيث تظهر معلومات محددة فقط.

مثلا لو عايزين نراقب الـ ICMP Traffic اللي جاي من R3 فقط، هنسخدم ACL نحدد فيها الـ ACL Traffic وبعددين نربطها بالـ Debug

```
R2(config)# access-list 100 permit icmp host 11.0.0.3 host 11.0.0.2
R2(config)# do debug ip packet detail 100
```

20.1.7. IPv6

ييدعم فقط الـ Extended Named ACL مع اختلاف بسيط في الـ Syntax

- استبدال ip access-list بـ ipv6 access-list •
- عند التطبيق تحت الـ Interface استخدام ipv6 traffic-filter access-list _name •

مثال

```
R2(config)# ipv6 access-list CISCO
R2(config-ipv6-acl)# permit ipv6 host 2001::1 host 2001::2
R2(config-ipv6-acl)# int f0/0
R2(config-if)# ipv6 traffic-filter CISCo in
```

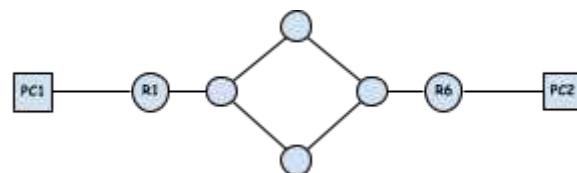
20.2. Notes

ملحوظة:

- اي ممكن نطبق عليه اربعة ACL فقط .. اتنين لـ IPv6 واحدة منهم In والثانية Out .. واتنين كمان لـ IPv6
- الـ Standard ACL بنحدد فيها Source فقط، وبالتالي الافضل هو تطبيقها قريب من الـ Destination
- اما الـ Extended ACL بنحدد فيها Source و بالتألي الافضل هو تطبيقها قريب من الـ Destination

مثال

- لو عايزين نمنع PC1 من الوصول لـ PC2
- لو حددت Standard ACL وطبقتها على R1 « كدا هتمنع اي Traffic يوصل من PC1 انه يعدي لاي شبكة او لاي Router .. اما لو طبقتها على R6 « الـ R6 هيوصل لـ PC2 وبعدين يقرر بعديه لـ PC2 ولا لا
 - لو حددت Extended ACL وطبقتها على R1 « الـ R1 الخاص بـ PC1 اللي رايح لـ PC2 هيتم منعه مباشرة، ودا الافضل .. اما لو طبقته على R6 هيمر على اكتر من Router وهيتم معالجته لحد ما يصل لـ R6 وبعدين يعمله Drop، ودا هيستهلك من موارد الشبكة على الفاضي



21. NAT

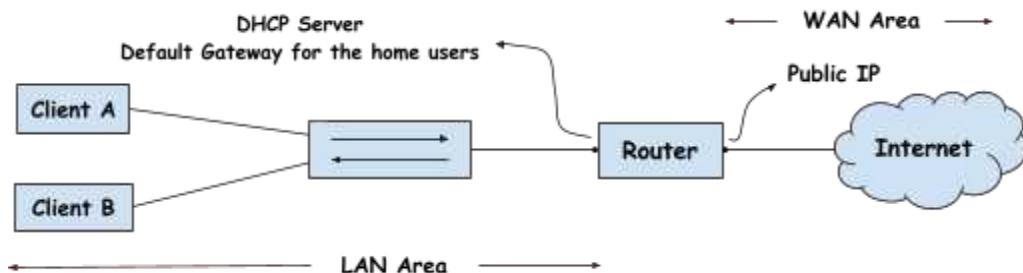
زي ما قلنا قبل كدا ان بسبب العدد المحدود لـ IPv4 وإنه مش هيكتفي كل الأجهزة اللي في العالم، كان لازم يكون في حل يسمح للأجهزة الداخلية في الشبكات إنها تتصل بالإنترنت بدون الحاجة لاستهلاك عدد كبير من الـ Public IPs. وهذا ظهر دور تقنية الـ NAT (اختصار لـ Network Address Translation) كأحد الحلول الأساسية على المدى القصير اللي يستخدم بجانب الـ Private IP، بحيث تسمح بعدد كبير من الأجهزة باستخدام Public IP واحد أو اكتر، وبالتالي توفر في العناوين المتاحة.

بالإضافة للتوفير في الـ IPv4، الـ NAT بتضيف طبقة أمان إضافية لأنها بتخفي العناوين الداخلية للشبكة عن الإنترت، وبتمنع الوصول المباشر للأجهزة الداخلية في الشبكة، ودا بيقلل فرص الهجمات الإلكترونية.

اما على المدى البعيد، ظهر بروتوكول IPv6 اللي بيكون من 128 بت، يعني بيتوفر عدد كبير جدا من الـ IPs. .NAT + Private IP مباشرة .. فكان الحل الأمثل هو استخدام IPv6 لـ IPv4 بس بسبب صعوبة الانتقال من IPv4 لـ IPv6 مباشرة..

الـ Private IP المحدد للـ Range

- Class A: 10.0.0.0/8
- Class B: 172.16.0.0 172.131.255.255/12
- Class C: 192.168.0.0 192.168.255.255/16



- كل Router على الـ Interface ييأخذ IP مختلف
- الـ Default IP يتحط في خانة الـ Default Gateway على الأجهزة المتصلة بالـ Router
- لأن دا البوابة اللي هتلطع الأجهزة للـ Internet وتوصيلهم بال شبكات الثانية.

- تقنية الـ NAT بتترجم كل IP في الـ Internet إلى Private Area .. وبالتالي ممكن نستخدم نفس الـ LAN في اي Private IPs في اي تانية .. ولو هنتواصل مع بعض عن طريق الـ Public IP اللي بيكون Unique على مستوى العالم.

- الـ Home Users يكون الـ Public IP بتاعهم مش ثابت .. يعني بيتغير كل فترة (يومين مثلاً) او كل ما تعمل Connect على الـ Router .. لأن الـ Router هيعمل Connect على الـ DHCP Server بتاع الـ ISP عشان ياخد IP جديد .. وفي حالة لو عايز IP ثابت .. لازم تشتري Static IP من الـ ISP

21.1. NAT Types

- **Static (one-to-one) Mapping**

- هنا كل Private IP بيتربط بـ Public IP ثابت، وده بيتم يدوياً.
- مفيد في حالة وجود سيرفرات داخلية (زي Mail Server أو Web Server) تحتاجة تبقى متاحة للوصول من الإنترنэт بعنوان ثابت.
- بيضيف طبقة حماية لأنه يخفى العناوين الداخلية، لكنه مش بيوفر في الـ IPv4 لأنه بيستهلك Public IP لكل جهاز.

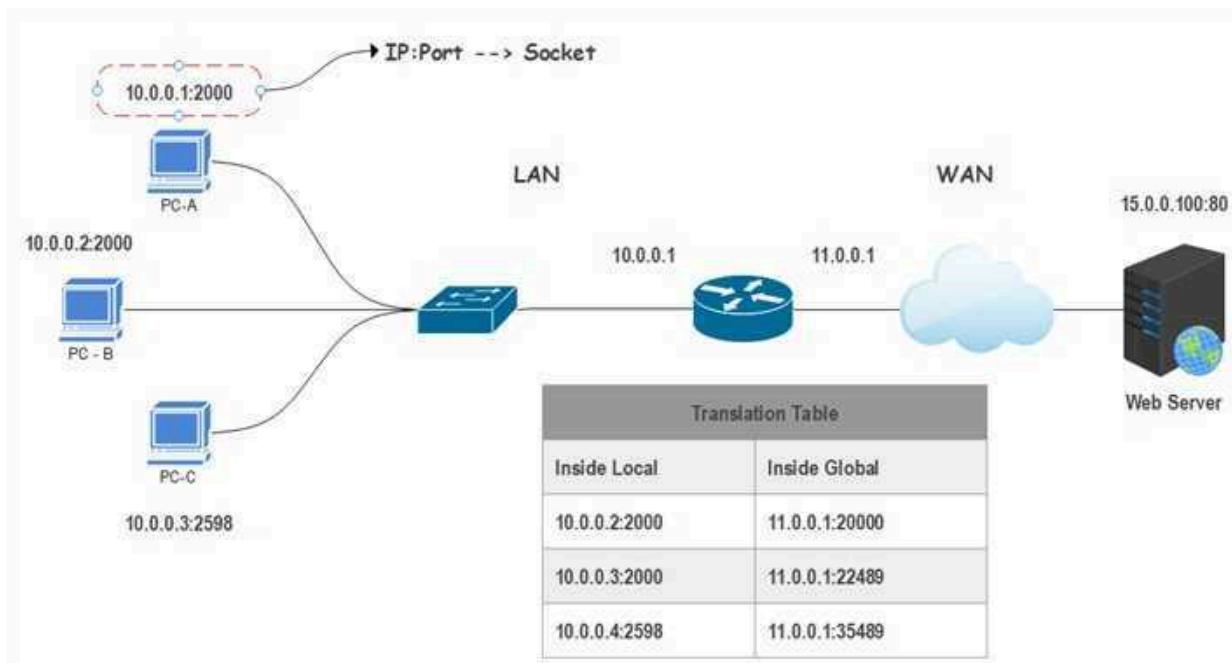
- **Dynamic**

مهمنته توفير مجموعة من الـ Public IPs وعندى LAN Network للـ LAN Network .. مثلاً عندى ثلاثة Public IP فال LAN خمس اجهزة » اي جهاز يطلع انترنت هيستخدم واحد من الـ Public IPs .. ولو التلاتة بقو مستخدمين وفي جهاز رابع عايز يطلع انترنت، لازم يستنى على ما واحد منهم ينهي اتصاله.

- **PAT (many-to-one) Mapping**

Port Address Translation او PAT وهو اشهر نوع، كل الأجهزة في الـ LAN بتستخدم Public IP واحد للوصول للإنترنэт واتسمى بالاسم دا لانه بيعمل Track لكل الـ Sessions لحد Layer 2 اللي فيها الـ Port Number، وبالتالي كل جهاز بيستخدم Port Number مختلف.

21.2. How PAT Work



- أي جهاز يطلب خدمة معينة، يستخدم Random Port + Private IP (من 1024 الى 65535).
- الـ Socket يطلق علىهم IP + Port Number.
- الراوتر يخزن الـ Socket اللي وصله في الـ Translation Table ويبدل الـ Public IP بالـ Source IP.
- ويستخدم Random Port معاهم، عادة اول Port بيكون نفس الـ Port اللي وصله. ودي الحالة الوحيدة اللي بيغير فيها الـ IP Header وهي لما يكون مطبق NAT.
- ممكن جهازين من نفس الـ LAN يستخدموا Random Port ويطلع نفس الـ Port، بس اهم حاجة ان الـ Socket مختلف .. لأن لو الـ Port طلع متطابق >> الـ IP مش هيتطابق.
- الـ Destination Server يفرق بين الـ Clients او الـ Packets اللي بنفس الـ IP عن طريق الـ Socket.

العملية دي اسمها NAT او SNAT، يعني الـ Router او الجهاز اللي بيعمل Natting بيبدل الـ Destination NAT او DNAT او Destination NAT جاي من خارج الشبكة على Source Address داخلي مثلًا (زي مثلاً لو عندي Web او Mail Server .. وهنالك Router بيبدل الـ Destination Server الى العنوان الحقيقي للـ Server داخل الشبكة).

عدد الأجهزة اللي يقدر يخدم عليها الـ Public IP الواحد

الراوتر اللي بيستخدم PAT نظرياً يقدر يخدم على 64512 جهاز لأن كل جهاز بيأخذ Port مختلف لما بيطلع للإنترنت، والـ Ports المتاحة بتبدأ من 1024 لحد 65535 (بما إن الـ Ports من 1 إلى 1023 مhogza للخدمات المعروفة زي HTTP, DNS, FTP).

في الواقع، العدد اللي يقدر الراوتر يخدمه بيعتمد على:

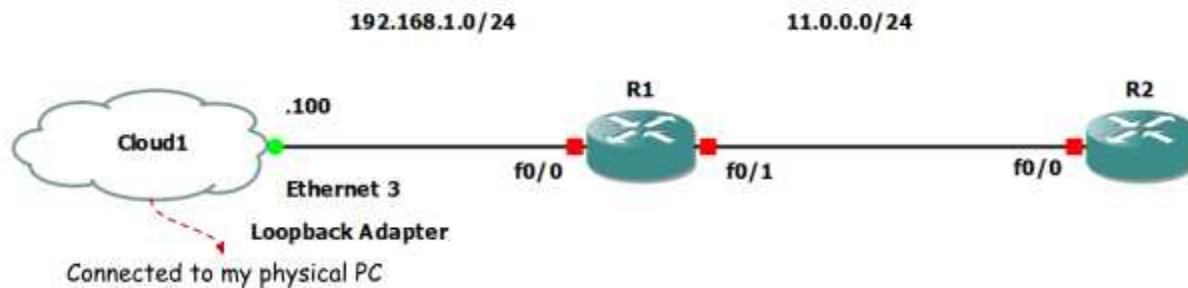
1. قدرة الهايبردويير بتاع الراوتر (RAM, CPU).
2. عدد الـ NAT Sessions اللي يقدر يخزنها في الـ Translation Table.
3. حجم الترافيك لكل جهاز (لو الأجهزة بتستهلك عدد كبير من الاتصالات، عدد المستخدمين هيقل). لأن كل جهاز بيستخدم Socket مختلف لكل خدمة بيطلبها. ومثلاً لو فتحت أي موقع، ممكن أي صورة أو فيديو او حاجة معينة تكون Hosted على موقع تاني .. وبالتالي بيقى ظاهر عندك انك فاتح موقع معين، لكن فالخلفية في اتصال مع كل موقع من الواقع دي عشان تقدر تفتح الصورة.

وتقدير تشووف كمية الـ Sessions المفتوحة عندك باستخدام الامر cmd

Proto	Local Address	Foreign Address	State
TCP	127.0.0.1:1025	127.0.0.1:1026	ESTABLISHED
TCP	127.0.0.1:1026	127.0.0.1:1025	ESTABLISHED
TCP	192.168.2.7:9487	149.154.167.91:80	ESTABLISHED
TCP	192.168.2.7:9495	149.154.167.91:80	ESTABLISHED
TCP	192.168.2.7:9519	102.132.103.8:443	ESTABLISHED
TCP	192.168.2.7:9522	102.132.103.59:443	ESTABLISHED
TCP	192.168.2.7:9523	104.16.102.112:443	ESTABLISHED
TCP	192.168.2.7:9532	104.16.102.112:443	ESTABLISHED
TCP	192.168.2.7:9536	104.16.102.112:443	ESTABLISHED
TCP	192.168.2.7:9578	104.16.102.112:443	ESTABLISHED
TCP	192.168.2.7:9604	20.199.120.85:443	ESTABLISHED
TCP	192.168.2.7:9637	20.199.120.85:443	ESTABLISHED

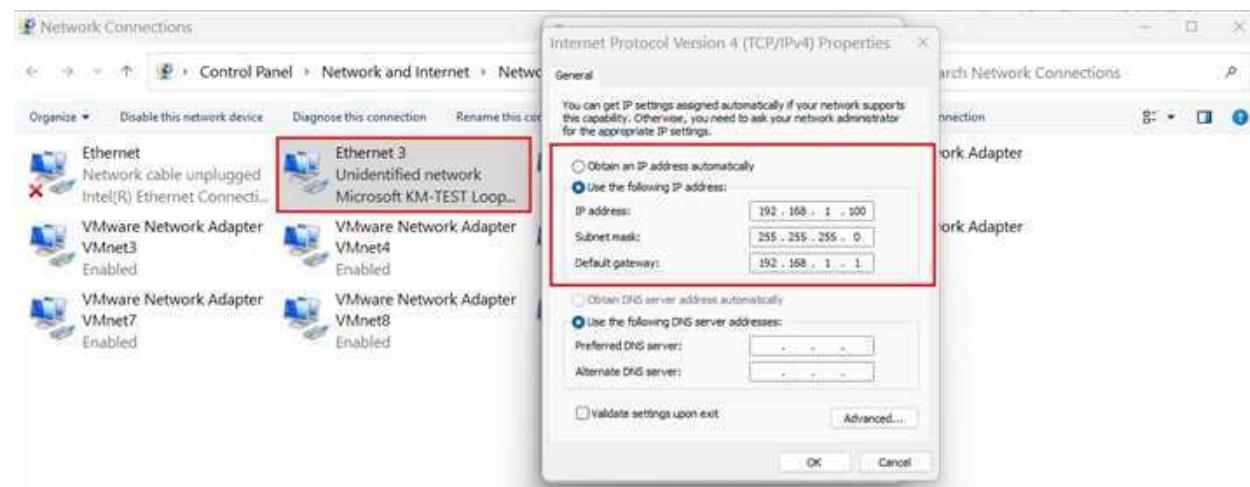
4. وبتعتمد على نوع الراوتر برضو ... وممكن تكون من 4000 الى 10000 جهاز

21.3. LAB Configuration



لو استخدمت فال Lab دي Router 3725 h او 3600 .. اول ما تطبق NAT هتلaci ال Router هنج وعمل يظهر Errors كتير .. والسبب ان الرامات بتاعته قليلة ومش بتتحمل >> وبالتالي ممكن تعلي الرامات بتاعته او تستخدم Router 7200

Configure IPs



```
R1(config)# int f0/0
R1(config-if)# ip add 192.168.1.1 255.255.255.0
R1(config-if)# no sh
R1(config-if)# int f0/1
R1(config-if)# ip add 11.0.0.1 255.255.255.0
R1(config-if)# no sh
```

```
R2(config)# int f0/0
R2(config-if)# ip add 11.0.0.2 255.255.255.0
R2(config-if)# no sh
```

Configure Static Routing

```
R2(config)# ip route 192.168.1.0 255.255.255.0 11.0.0.1
```

Test Connectivity

```
R2(config)#  
R2(config)# do ping 192.168.1.0  
Type escape sequence to abort.  
Sending 5, 100-byte ICMP Echos to 192.168.1.0, timeout is 2 seconds:  
!!!!!  
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/32/36 ms  
R2(config)#

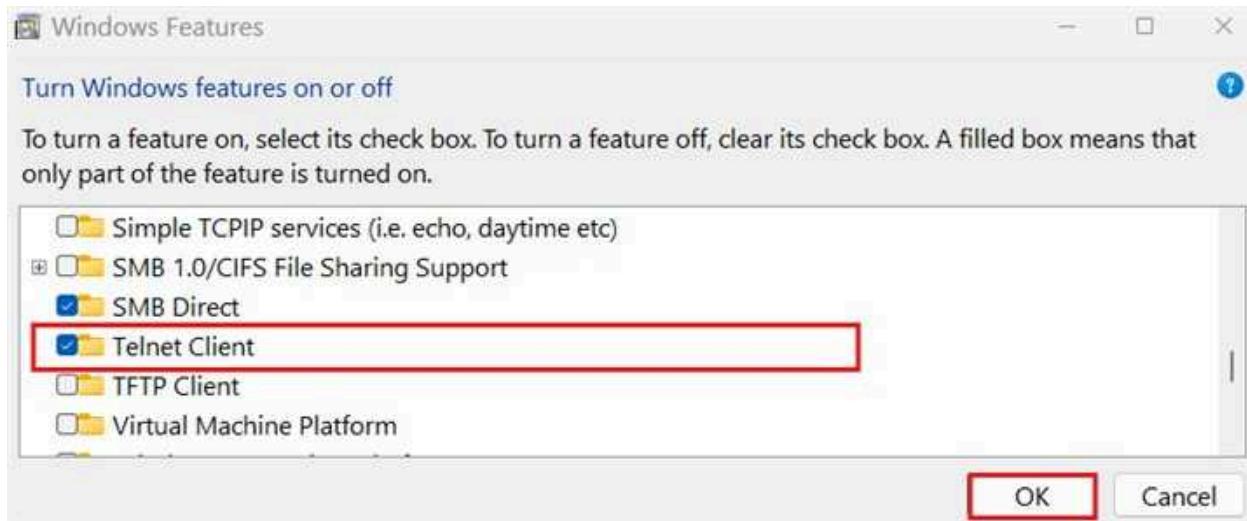
```

```
Command Prompt  
C:\Users\hp> ping 11.0.0.2  
Pinging 11.0.0.2 with 32 bytes of data:  
Reply from 11.0.0.2: bytes=32 time=42ms TTL=254  
Reply from 11.0.0.2: bytes=32 time=49ms TTL=254  
Reply from 11.0.0.2: bytes=32 time=48ms TTL=254  
Reply from 11.0.0.2: bytes=32 time=48ms TTL=254  
  
Ping statistics for 11.0.0.2:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
    Approximate round trip times in milli-seconds:  
        Minimum = 42ms, Maximum = 49ms, Average = 46ms  
  
C:\Users\hp>
```

Open Telnet on R2

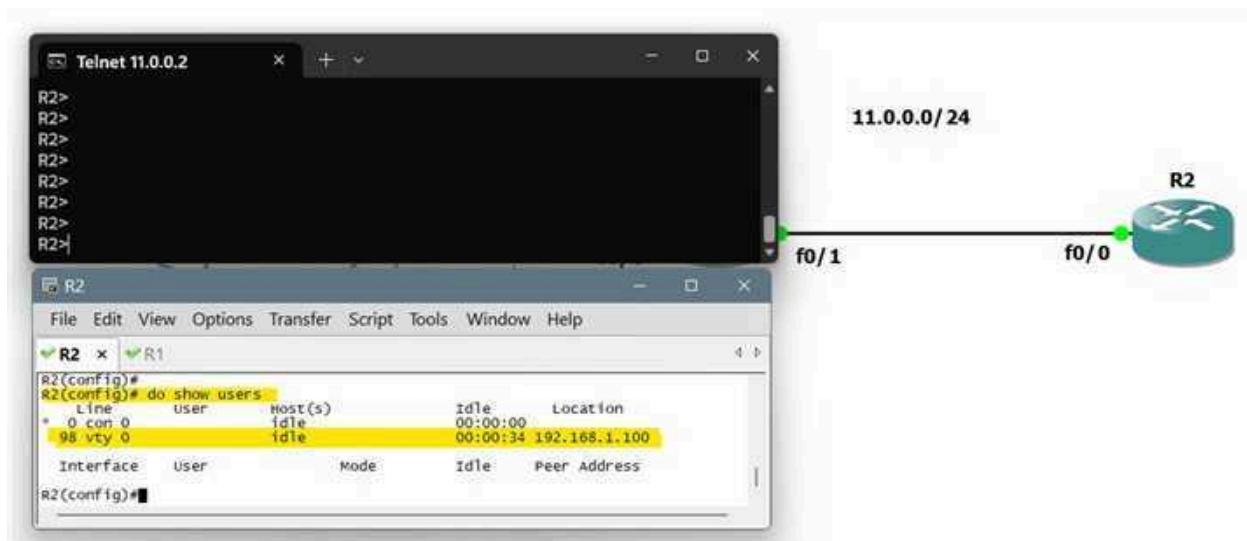
```
R2(config)# line vty 0 4  
R2(config-line)# no login
```

لاستخدام الـ Telnet على Windows 7 .. لازم تثبت الاول لأنها اشالت ابتداء من Windows Future



لو عملت Telnet على R2 وكتبت امر Show Users عشان تشوف مين اللي داخلين على الراوتر، هتلقي الـ

PC ظاهر بتاع الـ Private IP



اما بعد تطبيق الـ NAT .. لو عملت Telnet هيظهر الـ Public IP اللي حددته

21.3.1. Static NAT

Static NAT With Static Routing

لتطبيق Static NAT

- أول حاجة بنكتب NAT Rule شبيهة بال ACL بتحدد فيها الـ IP

- وبعدين بنطبقها تحت الـ Outside Interface والـ Inside Interface

نفترض اني اتعاقدت مع الـ ISP على Static IP 50.0.0.1 واداني

```
R1(config)# ip nat inside source static 192.168.1.100 50.0.0.1
R1(config)# int f0/0
R1(config-if)# ip nat inside
R1(config-if)# int f0/1
R1(config-if)# ip nat outside
```

بس احنا كنا مطبقين Static Routing من R2 على الـ IP القديم وبالتالي لازم نلفي الـ Route القديم ونضيف

على الـ IP الجديد Static Route

```
R2(config)# no ip route 192.168.1.0 255.255.255.0 11.0.0.1
R2(config)# ip route 50.0.0.0 255.255.255.0 11.0.0.1
```

بعد الـ IP الجديد .. لو عملت Telnet من الـ PC <> هيظهر بالـ Public IP

Line	User	Host(s)	Idle	Location
*	0 con 0	idle	00:00:00	
	2 vty 0	idle	00:00:09	50.0.0.1

عرض الـ Translated Packets على R1

```
R2# show ip nat translations
Pro Inside global           Inside local        Outside local       Outside global
icmp 50.0.0.1:3             192.168.1.100:3   11.0.0.2:3        11.0.0.2:3
tcp 50.0.0.1:43989          192.168.1.100:43989 11.0.0.2:23      11.0.0.2:23
--- 50.0.0.1                192.168.1.100      ---               ---
```

- الـ Global IP هو الـ Inside Global
- الـ Private IP هو الـ Inside Local
- الـ Public IP هو الـ Destination Outside Global
- الـ Destination Private IP هو الـ Destination Outside Local

في الوضع الطبيعي الرواوتر يبدل الـ Global IP Source بتعدي بالـ Global IP اللي حدده في الـ NAT، وبيسين الـ Destination زى ما هو .. اما في حالة لو الـ Destination اللي انا رايحله عامل NAT اصلاً >> هنا الرواوتر يبدل الـ Global IP Source للـ Destination Public IP للـ Destination IP بتعدي ويبدل الـ Destination Public IP بتعدي الـ Server بتعديهم غالباً اي موقع بيستخدم الـ Public IP

الـ Session بفضل موجودة لفترة محددة في الـ Translation Table وبعدين تتمسح .. وممكن تممسحها مباشرة عن طريق الأمر

```
R1(config)# clear ip nat translation *
```

الـ Static Routing عشان نطبق الـ Dynamic Routing مع Static NAT

Static NAT with Dynamic routing

```
R1(config)# router ospf 1
R1(config-router)# network 11.0.0.1 0.0.0.0 area 0
R1(config-router)# network 192.168.1.1 0.0.0.0 area 0
```

```
R2(config)# router ospf 1
R2(config-router)# network 11.0.0.2 0.0.0.0 area 0
```

المشكلة في الـ Dynamic Routing ان R2 مش هيشفوف الـ Public IP: 50.0.0.1 لانه مش موجود على Network 50.0.0.1 0.0.0.0 area 0 حتى لو كتبت امر `network 50.0.0.1 0.0.0.0 area 0` مش هيذاع لان مفيش Interface ييعمل Match مع الـ IP دا.

حل المشكلة دي <> هنعمل Loopback Interface ونحط عليه الـ IP دا

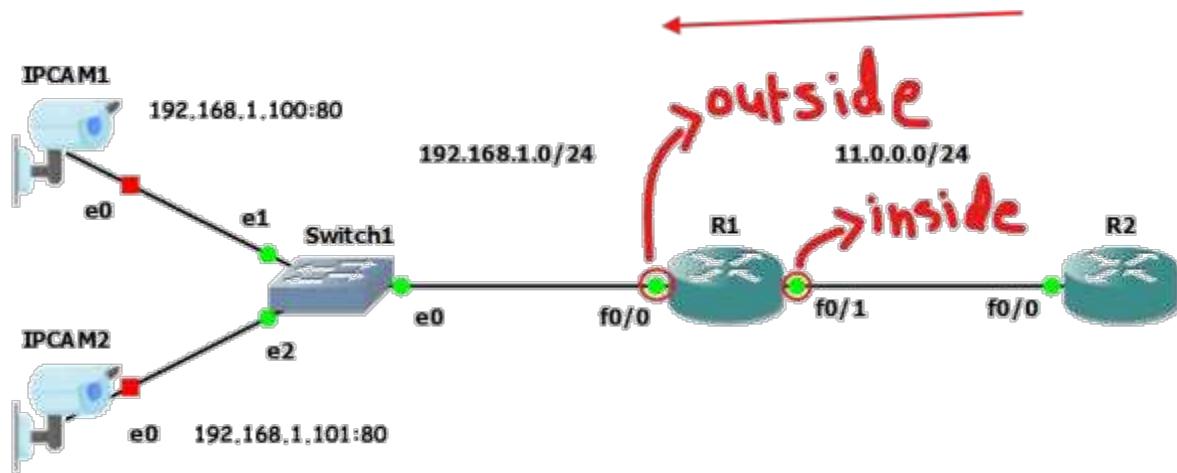
```
R1(config)# int loopback 1
R1(config-if)# ip add 50.0.0.1 255.255.255.0
R1(config-if)# router ospf 1
R1(config-router)# network 50.0.0.1 0.0.0.0 area 0
```

فايدة استخدام الـ Static NAT

لو عندي في الشركة او البيت كاميرات مراقبة .. الكاميرات دي بيبقى نوعها IP CAM يعني بتاخد IP عشان تقدر توصلها عن طريق الـ Browser مثلًا .. وبالتالي لو عندك اكتر من كاميرا هتضطر تشتري IP لكل واحدة عشان تقدر تدخل عليها من خارج الـ LAN بقاعدتك.

لكن في تانية اسمها Port Forwarding ودي بتمسح باستخدام Public IP واحد .. وبنعمل Configuration على الراوتر بحيث لو دخلت مثلاً على 50.0.0.1:2000 IP: يحولك على الكاميرا الاولى اللي الـ IP بتاعها داخل الـ LAN هو 192.168.1.100 .. ولو دخلت على 50.0.0.1:3000 IP: يدخلك على الثانية اللي واحدة عنوان 192.168.1.101 وهكذا.

واللي هيختلف هنا كمان، الـ Traffic هيكون جاي من برا على الـ LAN وبالتالي انا عايز مثلاً الـ Traffic اللي جاي على 50.0.0.1:2000 يتحول له 192.168.1.100:80 والـ Inside Interface فالحالة دي هيكون اللي من ناحية الـ WAN.



والـ Configuration هتكون بالشكل دا

```
R1(config)# ip nat inside source static tcp 192.168.1.100 80 50.0.0.1 2000
R1(config)# ip nat inside source static tcp 192.168.1.101 80 50.0.0.1 3000
```

الـ Remote Desktop دي ممكن اطبقها كمان في حالة وجود Server عايز اوصله من خلال Configuration وبالتالي هعمله Port Forwarding برضو.

21.3.2. Dynamic NAT

مثلا لو الـ ISP وفرلي الـ Range دا من الـ IPs:

50.0.0.1 to 50.0.0.3

اول حاجة بنعمل Pool بالـ IPs دي

```
R1(config)# ip nat pool WAN 50.0.0.1 50.0.0.3 netmask 255.255.255.0
```

وبعدين بنكتب Access-list نحدد فيها الـ Traffic اللي هنربطه بالـ NAT

```
R1(config)# access-list 1 permit 192.168.1.0 0.0.0.255
```

NAT Rule بـ Access-list

```
R1(config)# ip nat inside source list 1 pool WAN
```

طبق الـ NAT على الـ Interfaces

```
R1(config)# int f0/0
R1(config-if)# ip nat inside
R1(config-if)# int f0/1
R1(config-if)# ip ant outside
```

ملحوظة:

اول ما بتطبق الـ NAT .. الراوتر بيعمل Virtual Interface باسم NVO اختصار لـ Virtual Interface وبيأخذ الـ Private IP بتاع الراوتر.

على R1 ممكن تعمل Static Routing على شبكة الخمسينات عن طريق الامر:

```
R1(config)# ip route 50.0.0.0 255.255.255.0 11.0.0.1
```

الأمر التالي بيضيف Static Route للـ Public IPs دي

```
R1(config)# ip nat pool WAN 50.0.0.1 50.0.0.3 netmask 255.255.255.0 add-route
```

21.3.3. PAT

اول حاجة بنعمل Access List بحدد فيها الشبكة اللي هنطبق عليها PAT

```
R1(config)# access-list 1 permit 192.168.1.0 0.0.0.255
```

وبعدين بنكتب NAT Rule نحدد فيها ان الشبكة اللي في Access-list 1 هيحصلها Translation لـ IP بتابع الـ Interface اللي هتحددو.

```
R1(config)# ip nat inside source list 1 interface f0/1 overload
```

لو طبقت الامر دا من غير كلمة Overload « جهاز واحد فقط اللي هيقدر يستخدم الـ Public Interface .. اما مع استخدام الامر « كل Session هتسخدم نفس الـ IP مع Port مختلف.

وبعدين طبق الـ NAT على الـ Interfaces

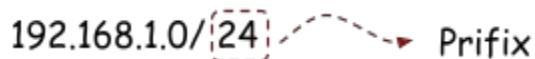
```
R1(config)# int f0/0
R1(config-if)# ip nat inside
R1(config-if)# int f0/1
R1(config-if)# ip nat outside
```

22. Router switching Modes

22.1. How Router Builds the Routing Table

- **Longest Prefix Match:**

الـ Router يختار المسار اللي فيه أطول Prefix بيتتطابق مع الـ Destination IP في الـ Packet



192.168.1.0/**24** → Prefix

- **Lowest Administrative Distance (AD)**

لو فيه أكثر من بروتوكول توجيهه بيحدد نفس الوجهة، الـ Router يختار المسار اللي له أقل AD

- **Lowest Metric**

لو فيه أكثر من مسار لنفس الوجهة وبنفس الـ AD، الـ Router يختار المسار اللي ليه أقل Metric

- **Equal Cost Load Balance**

لو فيه أكثر من مسار لنفس الوجهة وبنفس الـ Cost او الـ Metric . الـ Router بيوزن الـ Traffic

عليهم بالتساوي

22.2. Router Switching

الـ Router Switching او الـ Packet Switching هي الطريقة اللي بيتم من خلالها توجيه الـ Packet من الـ Interface Multilayer Switch لـ Router او الـ Multilayer Switch اللي بتخرج منه.

Process Switching

أول طريقة Switching سيسكو طورتها واستخدمتها، وهنا الـ Router بيعالج كل Packet بشكل منفصل عن طريق الـ CPU ومع كل Packet يراجع الـ Routing Table عشان يحدد المسار المناسب ليها .. والعملية دي بطينة وبتعمل Load على الـ CPU.

Fast Switching

بسبب بطء عملية الـ Process Switching قررو ان بدل ما الـ Router يعمل Lookup كامل على الـ Routing Table في كل مرة، بيتم تخزين النتائج دي في الـ Cache اسمه Fast Switching Cash بعد أول معالجة، ويتم استخدام نفس البيانات لتوجيه الـ Packets الثانية لنفس الوجهة بشكل أسرع.

أول Packet تدخل الراوتر ورایحة لـ Destination Network جديدة، الراوتر بيشوف لو الشبكة موجودة في الـ Cache.

- لو لقاها، بيعدل الـ Frame Header وبيعدت الـ Packet مباشرة لـ Interface المرتبط بالشبكة المطلوبة.

- لو الشبكة مش موجودة في الـ Cache، الراوتر بيعمل Lockup على الـ Routing Table عشان يحدد المسار المناسب وبعدين يخزن المعلومة دي في الـ Cache عشان يستخدمها لو في تانية رايحة نفس الشبكة بعد كده.

ممكن تحصر الطريقة دي في حمله Route Once And Switch

وفي طريقة تانية اسمها Net Flow Switch .. معتمدة على نفس فكرة الـ Fast Switch لكن بتستخدم أسرع Algorithm.

Cisco Express Forwarding - CEF

« Open Standard L3 Switching Mechanism هو Cisco Property .. وكان متقدم من Cisco ثم أصبح Cisco Express Forwarding عن طريق تقليل الضغط على الـ CPU .. وبالتالي آداء أعلى وأسرع.

الطريقة دي بتعتمد على استخدام وحدة معالجة خاصة بجانب الـ CPU اسمها ASIC وهي اختصار لـ Application Service Integrated Circuit الـ CEF بيعتمد على جزئين رئيسيين:

- الـ FIB اختصار لـ Forwarding Information Base وهي قاعدة بيانات بتحفظ مسارات الشبكة وبتكون محدثة حسب الـ Routing Table
 - الـ Adjacency Table ويكون فيه معلومات عن الـ Layer 2 عن الـ Direct Connected Devices وبجمع المعلومات من اكتر من مصدر زي الـ ARP
- ميكانزم الـ CEF شغال By Default على اجهزة Cisco .. ولو عملتلو « Disable Fast Switch » هيشتغل .Process Switch

بعض الأوامر

To enable fast-switching

```
Router# conf t  
Router(config)# int f1/0  
Router(config-if)# ip route-cache
```

To enable CEF

```
Router(config)# ip cef
```

To check FIB entries

```
Router# show ip cef
```

```
Router# show ip cef exact-route 1.1.1.1 3.3.3.3 dest-port 80
```

الأمر ده بيعرض المسار اللي الرووتر هيستخدمه عشان يوجه الـ Packets من عنوان IP 1.1.1.1 لعنوان 3.3.3.3 على بورت 80 باستخدام تقنية CEF.

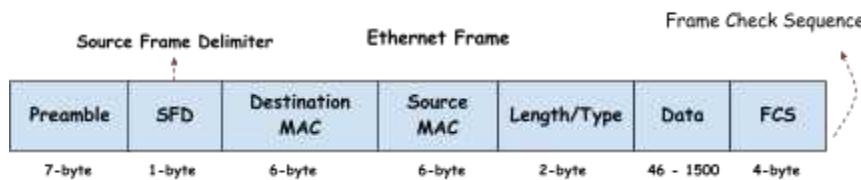
To show adjacency table details

```
Router# show adjacency
```

لعرض استخدام الـ CPU وتفاصيل العمليات اللي شغالة حالياً واستهلاكها للمعالج

```
Router# show process cpu
```

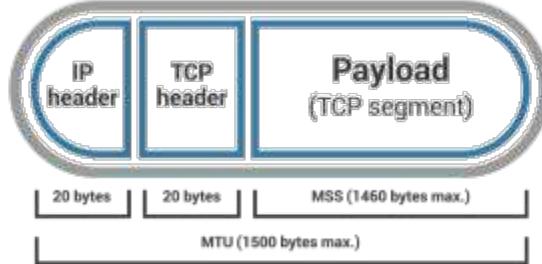
23. Switching Essentials



البيانات بتتجهز في الـ Application Layer بناء على البروتوكول المستخدم، وبعدين تنزل للـ Transport Layer

- في الـ Transport Layer بيعملها Segmentation ويتحدد حجم الـ Segment بناء على الـ Negotiation اللي يحصل في عملية الـ TCP Handshake، والمثال الشائع بيبقى 1460 بايت للـ Maximum Segment Size - MSS، حجم الـ Payload اللي بيتحدد هو الـ TCP Header .. وبعدين تتغلف في TCP او UDP Header .. مثل الـ TCP هيزود 20 بايت وهيبقى حجمها 1480 بايت.
- وبعدين تنزل على الـ Network Layer وتتغلف في IP Header ويبقى حجمها 1500 بايت وهيبقى اسمها .MTU - Maximum Transfer Unit

Data packet



$$MTU = (IP Header + TCP/UDP Header) + MSS$$

- وبعدين تنزل للـ Data Link Layer وتتغلف في Header وTrailer وحجمها يصل لـ 1518 بايت.
- الفرق بين الـ MTU والـ MSS .. ان لو الـ Packet تعدت الـ MTU الخاص بالجهاز، يتم تقسيمها لوحدات أصغر (والعملية دي اسمها Fragmentation .. أما لو تعدت الـ MSS «Drop» هيحصلها).

في السويفتش، أي قيمة بتتخزن في الـ MAC Address Table بفضل موجودة لفترة اسمها Aging Time واللي بتكون 300 ثانية بشكل افتراضي.

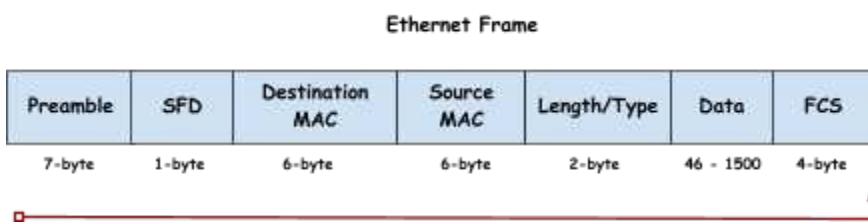
لو الجهاز بعث أو استقبل رسالة خلال المدة دي، الـ Aging Time بيتحدد تاني. أما لو محصلش إرسال أو استقبال خلال مدة الـ Aging Time، يتم إزالة الـ Entry تلقائياً.

وزي ما عرفنا في طريقة عمل السويفتش ان أول Frame بيحصله Flood، بس دا بيحصل في حالة ان مفيش خاص بالـ MAC Address Table دا في الـ Destination Entry .. أما لو في الـ Destination Entry بيتمن الإرسال مباشرة من غير Frame لأول Flood.

23.0.1. Switching Modes

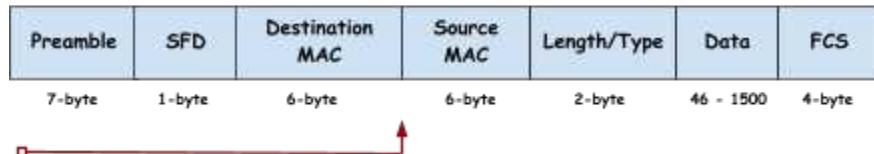
- Store and Forward
- Cut Through
- Fragment Free

كل الـ Switch على الـ Interface Buffer بيبقى له جزء محجوز من الـ RAM اسمه RAM •
لو الـ Switch شغال على وضع Store & Forward وده النوع الأكثر شيوعاً •



- البيانات بتوصل على هيئة Zeros & Ones و بتتخزن في الـ Buffer لحد ما الـ Frame يوصل بحجمه الكامل (Byte 1518).
- وبعدين بيعمل للبيانات دي Check باستخدام الـ CRC او FCS Technique.
- لو البيانات مش سليمة او فيها تعديل او Created، بيعملها Drop من السويفتش .. أما لو سليمة، بيدأ السويفتش بيعتها بناء على الـ Destination MAC

Ethernet Frame



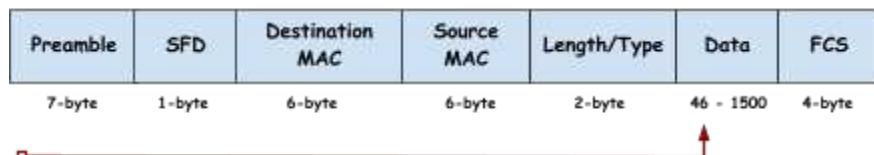
لو السويتش شغال على وضع Cut Through •

البيانات بتخزن في Buffer لحد ما السويتش يصل لل Destination MAC Address ○

وبعدين بيبدأ بيعت ال Bits اللي بعد كدا لل Dest مباشرة بدون انتظار.

النوع دا بيستخدم في الشبكات اللي محتاجة Low Latency ○

Ethernet Frame



لو السويتش شغال على وضع Fragment Free •

السويفت بيستقبل أول 64 بايت من ال Frame ويتأكد إنها سليمة قبل ما يبدأ إرسالها .Bit By Bit

لو ال Frame حجمه أقل من 64 Byte، بنسميه Runt يعني "قزم" وبيكون Corrupted .. أما لو حجمه أكبر من Giant، بنسميه Byte 1518

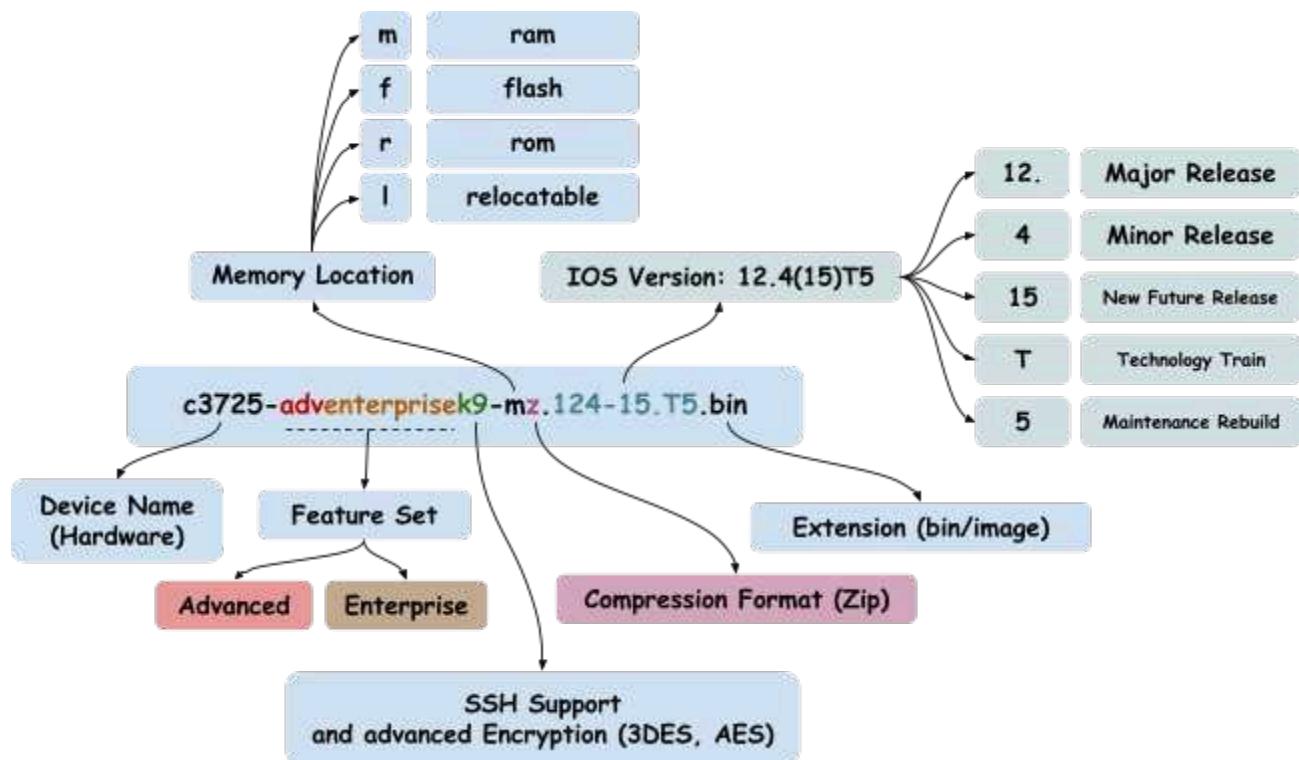
24. IOS License

24.1. CISCO IOS Naming Conventions

أسماء إصدارات Cisco IOS تتبع نظام معين عشان توضح الإصدار، الميزات، والنوع الخاص بـ IOS اللي شغال على الجهاز. كل اسم لإصدار IOS بيكون مكون من عدة أجزاء، وكل جزء ليه معنى محدد.

الصيغة العامة لاسم IOS:

c3725-adventerprisek9-mz.124-15.T5.bin



• في الـ Memory Location بيبقى في حرف بيوضح مكان تنفيذ الـ .. Image

◦ مثلًا m يعني الـ Image بتعملها Load على الـ RAM عشان تشتعل

◦ او ممكن يتعملها Run من الـ Flash

◦ او من الـ ROM

• خانة الـ **Feature Sets** بتدل على الـ **Image License** اللي بتسخدمها او الـ **ipbase** .. مثلا:

- إصدار أساسى مع وظائف IP بسيطة_<**ipbase**>
- إصدار متقدم للأمان مع دعم VPN والتشفير_<**advancedsecurityk9**>
- إصدار متكامل يشمل كل الميزات المتاحة_<**adventureprisek9**>
- إصدار مخصص للـ **Service Providers** <**spservices**>
- في اصدار IOS 15 هتلaci اسم النسخة **Universalk9**

• الـ **IOS Version** عبارة عن رقم الإصدار الرئيسي (**Main or Major Version**) اللي هو 12.2 او 12.4

او 15.2 وآخر اصدار نزل هو 15.7 والاصدار دا لما بيزيid بنسميه **Technology Train** يعني في تغير كبير حصل في النسخة دي.

◦ بعد رقم الإصدار الرئيسي بيكون في رقم بين قوسين اسمه **Throttle** وهو عبارة عن **Minor Version Number** بيدل على ان في بعض المميزات الجديدة او حل بعض المشاكل **Bug Fix** .. او دعم لأوامر جديدة.

◦ حرف **T** معناه إصدار خاص بالدعم التقني والتحديثات السريعة , وبالتالي ممكن يبقى فيه تحديثات اكتر و **Bugs** اكتر .. وممكن تلaci حرف **M** بدل **T** ودا معناه ان الاصدار دا مستقر .. وب Russo بيكون في **Special Releases** زي (**S, E, XB.. etc**) ودي بتبقى إصدارات متخصصة لنماذج معينة من الأجهزة أو لاستخدامات محددة.

◦ الرقم اللي بعد الـ **T** عبارة عن تحديث بسيط فيه **Bug Fix Only**.

ملاحظة:

يفضل التحقق من الـ MD5 Hash بعد تنزيل تحديث معين للتأكد من سلامة الملف وعدم تعرضه للتتعديل أو التلف أثناء التنزيل وبالخصوص لو منزل التحديث من مصدر غير رسمي.

للتحقق من الـ MD5 Hash

```
switch# verify /md5 <Location>:<IOS_filename> <expected_MD5_hash>
```

- <location>: مكان الملف (مثل flash:, usb0:, أو .:bootflash:)
- <filename>: اسم ملف IOS (مثل c2960-lanbasek9-mz.150-2.SE4.bin)
- <expected_MD5_hash>: قيمة MD5 الموجودة على موقع CISCO

مثال:

```
switch# verify /md5 flash0:c2960-lanbasek9-mz.150-2.SE4.bin  
1a2b3c4d5e6f7g8h9i0j1k2l3m4n5o6p
```

```
.....  
.....  
.....  
.....  
..... MD5 of  
flash0:c2960-lanbasek9-mz.150-2.SE4.bin Done!  
Verified (flash:c2960-lanbasek9-mz.150-2.SE4.bin) =  
1a2b3c4d5e6f7g8h9i0j1k2l3m4n5o6p.
```

لو النسخة متطابقة هيقولك Verified

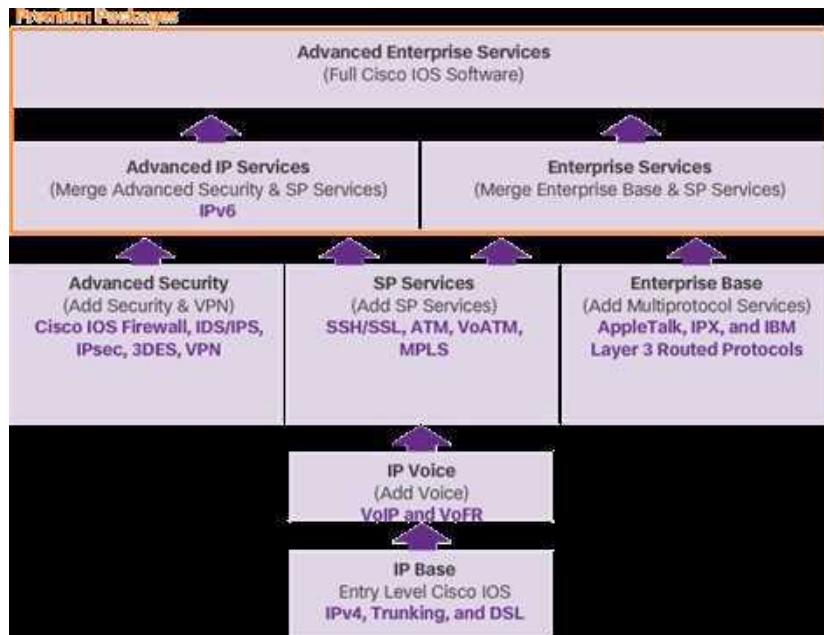
لو مش متطابقة هيطلع Error

```
switch# verify /md5 flash0:c2960-lanbasek9-mz.150-2.SE4.bin  
1a2b3c4d5e6f7g8h9i0j1k2l3m4n5o6p  
.....<output truncated>.....Done!  
%Error verifying flash0:c2960-lanbasek9-mz.150-2.SE4.bin  
Computed signature = x1y2z3a4b5c6d7e8f9g0h1i2j3k4l5m6 Submitted  
signature = 1a2b3c4d5e6f7g8h9i0j1k2l3m4n5o6p
```

ممكن تكتب الامر بدون قيمة الا MD5 عشان يعمل MD5 Hash وبعدين تقدر تقارنه في وقت تاني

24.2. CISCO IOS System Image Packages

قبل إصدار 15.0 من Cisco IOS، كان في 8 حزم برمجية (Images) مختلفة متاحة لأجهزة الراوتر من Cisco كل حزمة كان فيها على مميزات محددة بناء على احتياجات الشبكة.



1. حزمة IP Base - ipbase دي الحزمة الأساسية اللي فيها الوظائف العادي زي بروتوكولات التوجيه.
2. حزمة IP Voice - ipvoice بتضييف دعم لميزات .VoIP
3. حزمة Enterprise Base - entbase فيها ميزات متقدمة للشبكات في المؤسسات الكبيرة.
4. حزمة QoS and MPLS Service Providers - spservices موجهة لـ Service Providers و بتدعم مميزات QoS و MPLS.
5. حزمة Advanced Security - advsecurity بتضييف ميزات الحماية زي VPN, Firewall, AAA.
6. حزمة Advanced IP Services - advipservices بتدعم ميزات متقدمة زي IPv6, MPLS, Layer 3.
7. حزمة Enterprise Services - entservices فيها ميزات أكثر من Base، زي دعم بروتوكولات التوجيه ودي جموعها حزم رقم 3 و 4 و 5.

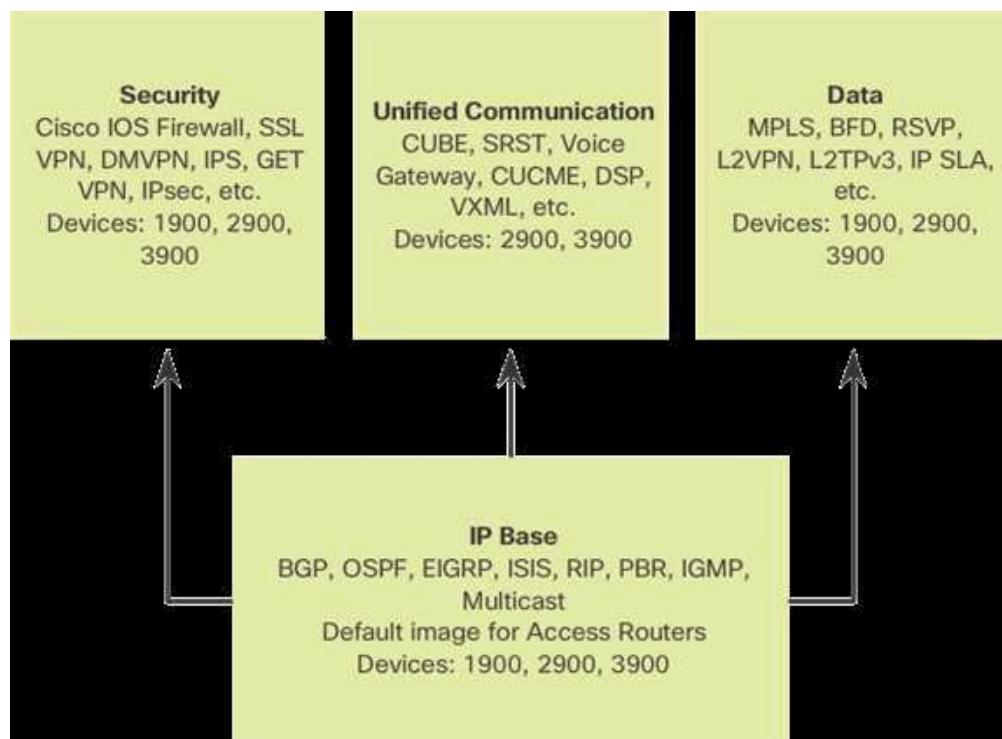
.8 حزمة Advanced Enterprise Services - adventerpriseservices فيها أكبر وأشمل حزمة،

كل الميزات تقريباً. وجمهو فيها حزمة رقم 6 و 7

بعد إصدار IOS 15.0، سيسكو بسطت الموضوع وخلت كل الميزات موجودة في Image واحدة، والفرق بقى في نوع الترخيص (License-Based Model). واسمها Single Universal Image with built-in (.Futures .. فبدل ما كنت بتشتري نظام كامل عشان تستخدم معينة «Technology» Train واحد فيه كل الميزات وانتا مجرد بتشتري Future License File عشان تفتح الـ اللي انتا تحتاجها

24.3. IOS 15 System Image Packages

سيسكو اعتمدت نظام تراخيص جديد اسمه Cisco Universal Image Licensing، واللي بيستغل على مبدأ إن كل الراوترات والـ Switches بيكون فيها نفس الـ Software Licensing، لكن الـ هي اللي بتحدد إيه الميزات اللي تقدر تستخدمها. أهم التراخيص المتاحة:



- الترخيص الأساسي اللي بييجي افتراضي على معظم الأجهزة هو IP Base، ويشمل الميزات الأساسية زي التوجيه العادي (RIP, OSPF, EIGRP) والـ NAT.

- لاحظ ان الـ Universal License مدعومة على Generation 2 من الـ Network Devices اللي هي

Series 900

الجدول التالي بيوضح كل License على اليمين تقابل ايه من الـ Images اللي كانت موجودة قبل Cisco IOS

15.X

Reformation Packaging	Suggested Transition to Simplified Packaging
IP Base	IP Base
IP Voice	Unified Communications
Enterprise Base	Data
Enterprise Services	Data + Unified Communications
SP Services	Data + Unified Communications (for feature parity and Enterprise features)
Advanced Security	Security + Unified Communications + Data (for feature parity and Enterprise features)
Advanced IP Services	Security + + Unified Communications + Data (for feature parity and Enterprise features)
Advanced Enterprise Services	Security + Unified Communications + Data

24.4. Managing CISCO IOS

نقل نسخة من النظام على TFTP Server من خلال الامر

```
switch# copy flash0://c2960-lanbasek9-mz.150-2.SE4.bin tftp
```

لعمل Boot من الـ Flash

```
switch# boot system flash0://c2960-lanbasek9-mz.150-2.SE4.bin
```

او من TFTP Server

```
switch# boot system tftp://c2960-lanbasek9-mz.150-2.SE4.bin
```

او من الـ ROM عشان تدخل على الـ Bootstrap

```
switch# boot system rom
```

24.5. Software License

لو عايز تشتري **License** معينة لجهاز **Cisco** سواء سويتش أو راوتر .. ممكن وانتا بتشتري الراوترات من **Cisco** مثلاً تطلب منهم يفعلو **License** معينة، وبالتالي الاجهزه هتتيجي مفعله بالـ **License** اللي انتا تحتاجها .. او في حالة لو عندك الاجهزه اصلاً ممكن لو عميل كبير (شركة كبيرة او مؤسسه)، بتتعامل مع فريق مبيعات **Cisco** مباشرةً او شريك معتمد كبير، وعادةً بيبقى فيه عقد او اتفاقية. وبيوفروا لك تراخيص ودعم فني. لو عميل صغير بتروح لموزع معتمد من **Cisco** في مصر او في البلد اللي انتا فيها (وبيبقى اسمه **Reseller**) او تشتري اونلاين من موقع **Cisco** الرسمي لو متاح.

1. الشراء

بتحدد نوع الترخيص اللي عايزه (زي **Advanced IP Services**) حسب احتياجاتك.

- هيدولك ملف الترخيص (امتداده **.lic**) مع كود تفعيل او - **Product Authorization Key** أو **PAK** وهو كود او رقم بتاخده لما تشتري الترخيص من **Cisco** او من الـ **Reseller**. بيبقى زي "فاتورة" او "تذكرة" بتثبت إنك دفعت فلوس عشان الترخيص ده.

والـ **PAK** ممكن يصل على الايميل بالشكل دا



- بتستخدم الـ **PAK** ده على موقع **Cisco** الرسمي (**Cisco Software Licensing Portal**) للجهاز بتاعك. او من خلال **Cisco License** عشان تطلب ملف الترخيص (امتداده **.lic**) للجهاز بتاعك.

لو هتتعامل مع اجهزة كتير وهو Cisco من Free Software Manager - CLM

www.cisco.com/go/clm

○ بتدخل الـ PAK مع الـ UDI الخاص بالجهاز في الموقع، وبعدين Cisco بتديك ملف الترخيص

اللي هتسخدمه في التفعيل. اختصار UDI هو رقم Unique Device Identifier

لكل جهاز من CISCO

للتحقق من الـ UDI الخاص بالجهاز

switch# show license udi			
Device#	PID	SN	UDI
*1	CISC02911/K9	JAE1234ABCD	CISC02911/K9:JAE1234ABCD

يتكون من جزئين:

.(C2960X-24TS-L: نوع الجهاز (زي PID - Product ID ■

.SN: الرقم التسلسلي الخاص بالجهاز ■

○ اكتب في المتصفح www.cisco.com/go/lisence واعمل Login وبعدين اختار Traditional

Lisence



ممكن تضغط على Watch Now او اضغط على [PAK](#) عشان تشو夫 ازاي تجيب الـ PAK
to license registration portal

اضغط على From a new PAK وبعدين هيطلب منك تكتب الـ PAK و الـ UDI وبعدين بتحمل ملف الـ License وتنقله للفلاش بتاع الجهاز او على TFTP Server او على اي مكان بالنسبة للجهاز .. اما لو مش معاك PAK ممكن تجرب Demo عن طريق الضغط Evaluation على Shared Account بس لازم بتاعك يكون بيدعم الـ Demo and evaluation License

2. التفعيل:

- انقل ملف الـ License على فلاشة او على TFTP Server

```
switch# copy tftp flash
Address or name of remote host []? 10.0.0.1
Source filename []? FHH12345678.lic
Destination filename []? FHH12345678.lic
```

- تفعيل الترخيص من خلال كتابة الأمر

```
switch# license install flash:FHH12345678.lic
```

- او ممكن تعمله Install مباشرة من الا TFTP من خلال الامر

```
switch# license install tftp://10.0.0.1/license/5400/38a.lic
```

لو كل حاجة تمام، هيظهر رسالة إن الترخيص اتفّعل.

- وبعدين اعمل Reload عن طريق الامر Router لـ Reload ولو بتستخدم النسخة الـ Evaluation فمش هتحتاج تعامل Reload وهاخد 60 يوم مجانا وبعد كدا لازم تشتري

3. للتحقق:

لعرض التراخيص المتوفرة وحالة كل ترخيص

```
switch# show license feature
```

Feature name	Enforcement	Evaluation	Subscription	Enabled	RightToUse
ipbase	no	no	no	yes	no
securityk9	yes	yes	no	no	yes
datak9	yes	no	no	no	yes
ucl9	yes	yes	no	no	yes

اسم الميزة (زي IPBase, Security, Data)	Feature Name
--	--------------

هل الميزة دي تحتاج ترخيص عشان تشغّل؟	Enforcement
--------------------------------------	-------------

هل الميزة دي ممكن تتجرب بترخيص تجريبي؟	Evaluation
--	------------

هل الميزة دي بتحتاج اشتراك شهري/سنوي؟ = اشتراك مطلوب, no = لا تحتاج اشتراك (yes)	Subscription
--	--------------

هل الميزة مفعّلة حالياً على الجهاز؟	Enabled
-------------------------------------	---------

هل مسموح باستخدام الميزة حتى لو الترخيص مش موجود رسمي؟ RightToUse - RTU

(Evaluation) = مسموح, no = غير مسموح (yes)

Evaluation License تفعيل

```
switch(config)# license boot module c2900 technology-package securityk9
```

وبعدين اعمل Accept واحفظ الـ Configuration عن طريق كتابة الامر wr ثم

عرض تفاصيل عن حالة كل ترخيص

```
switch# show license all
```

or show license detail

امر show version يعرض معلومات عن التراخيص في اصدارات IOS 15

يفضل تعمل Backup لملف الـ License.lic عن طريق الامر save license flash0:all_ License.lic

لعمل Uninstall للترخيص <> اكتب الامر

Disable the technology package

```
switch(config)# license boot module c2900 technology-package securityk9  
disable  
switch(config)# do reload
```

Clear the license

```
switch# license Clear securityk9  
switch# conf t  
switch(config)# no license boot module c2900 technology-package securityk9  
disable  
switch(config)# do reload
```

25. Virtual LAN - VLAN

الـ Vlan عبارة عن تقنية بتسمح بتقسيم السويفتش الى مجموعة من الشبكات الوهمية، وبالتالي ممكن نخلify كل مجموعة من الـ Interfaces في شبكة معينة.

السويفتش By Default بيكون عليه شبكة وهمية واحدة اسمها VLAN1، وكل الـ Interfaces بتكون فيها وبالتالي بيقدرو يشوفو بعض.

ليه بنستخدم الـ Vlan

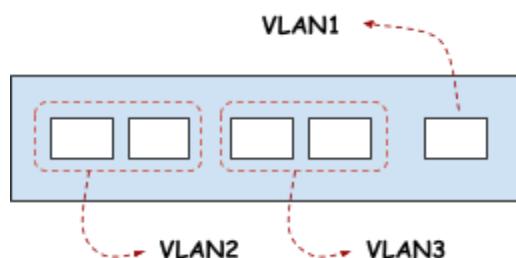
- تقليل حجم الـ Broadcast Domain .. لأن كل شبكة او Vlan بتكون في Broadcast Domain ..
- لوحدها.

- زيادة الأمان .. لأن لو حصل اختراق في شبكة معينة، مثلاً «vlan2» مش هيوصل لباقي الشبكات.

مثال:

لو عندي في الشركة 3 أقسام، وهم (IT - HR - Sales) .. وعايز كل جهاز يصل للأجهزة اللي معاه في نفس IP القسم بس، ممكن اخلي كل قسم لوحده عن طريق الـ Network ID .. ولكن لو أحد الـ Users قدر يغير الـ IP بتاعه «» ممكن يقدر يصل لقسم تاني.

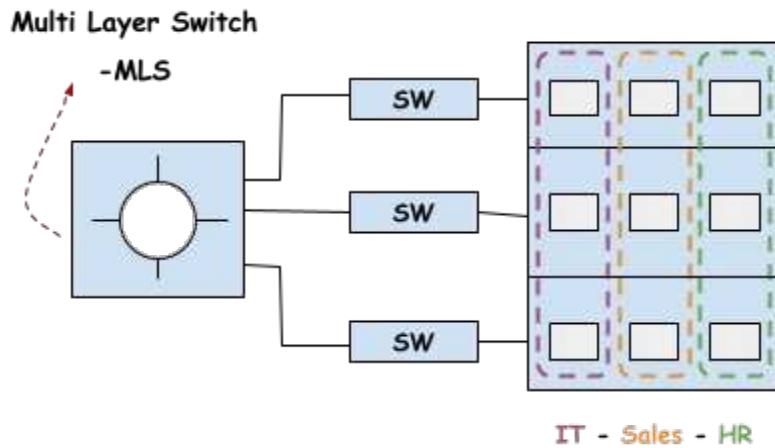
أما لو فصلت كل قسم عن طريق VLAN مختلفة .. مفيش مستخدم هيقدر يغير الـ Vlan اللي هوا فيها، لنه مش هيبقى عارف الـ Vlan اللي هو فيها اصلاً .. وكمان الـ Vlan Configuration يتم على السويفتش.



في الصورة مثال على عمل اتنين Vlans .. ووضع كل اتنين Vlan في Interface مختلفه .. وفي موجود في الـ Vlan الرئيسي اللي هي VLAN1.

مثلا لو حذفت VLAN 3 واللي فيها مجموعة من الـ Interfaces ايه اللي هيحصل !

- الـ Interfaces مش هتنقل للـ VLAN الاافتراضية (VLAN1) ولكن هتكون Homeless، ولون البورت هيكون Amper يعني برتقالي مایل للاحمر.
- عشان ارجع البورت « لازم اعمل Configure لنفس الـ VLAN تاني .. او انقلها لـ VLAN 1 عن طريق الاوامر.



ممكن نفس الـ VLAN تتعمل على اكتر من سويتش عشان الاجهزه اللي عليهم تقدر تتواصل مع بعض .. وعشان الـ VLANs المختلفة توصل لبعضها « لازم نعمل Routing بينهم عن طريق Switch بيفهم الـ Routing وهو الـ MLS

25.1. VLAN ID

كل VLAN يبيت اضافتها بيكون لها اسم ورقم مميز اسمه VLAN ID - VID

- الـ VLAN ID عبارة عن 12 بت .. يعني عندنا $2^{12} = 4096$ VLAN او من 0 الى 4095
- كل VLAN ممكن تاخذ اسم .. زي مثلا VLAN 2 ممكن تاخذ اسم HR
- لو خليت الـ VLAN من غير تسمية « هتاخذ اسم "VLAN" متبوع برقم الـ VLAN ID » متمثل في اربعة Digit

مثال:

VLAN 10 → VLAN0010

25.2. VLAN Ranges

- **Normal Range (Standard Range from IEEE)**

دا الـ Standard Range والي مدحوم على كل الأجهزة القديمة او الحديثة

- VLAN 1 (Default Vlan)

هي الـ Vlan الافتراضية « كل الـ Interfaces بتكون فيها ولا يمكن حذفها او اعادة تسميتها

- From VID 2 to 1001
- From VID 1002 to 1005

محجوزين لشبكات الـ FDDI والـ Token Ring وبرضو متقدرش تحذفهم او تعمل اعادة تسمية

- **Extended**

- From VID 1006 to 4094

عادة بيتم استخدامهم مع شبكات الـ Service Providers

- VLAN 4095

مش كل الأجهزة ولا كل الـ Vendors بيعدمو Vlan 4095

عموماً Vlan 0 لهم استخدامات خاصة

25.3. Types of Vlan Membership

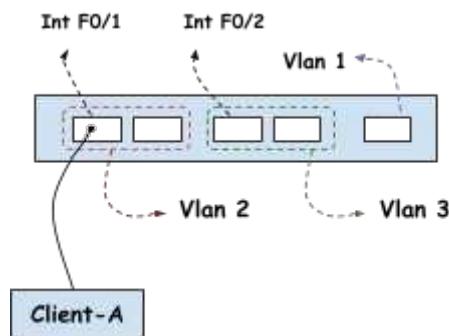
في طريقتين لعمل لـ Vlans Assaigne

Static Vlan

في الـ Static Method هي Port-based method يعني بتعتمد على وضع البورت نفسه في Vlan معينة

بطريقة Manual .. وبالتالي مثل:

- لو عندي Int F0/1 موجود في 2 .. Vlan 3 موجود في Int F0/2 .. و Client-A متصل بـ .Int F0/1



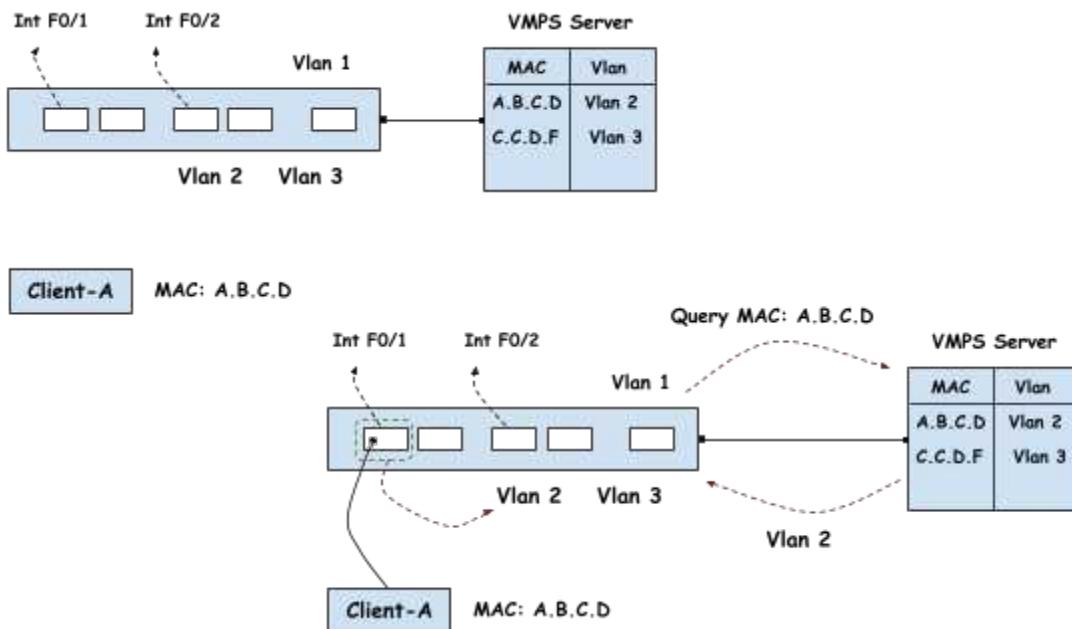
- لو شلت الـ Cable الخاص بـ Client-A ووصلته بـ Client-A من Interface F0/2 ممكن اغير الـ Configuration المطبقة على الـ Vlan 3.

- وبالتالي عشان اغير الـ Vlan بتاعت Client-A ممكن اغير الـ Configuration المطبقة على الـ Client-A او اشيل الـ Cable بتاع Client-A ووصله بـ Interface F0/1 الموجود في الـ Vlan اللي انا عايزها.

Dynamic VLAN

في الطريقة الـ **Dynamic Assign** للبورت في Switch معينة بناء على معلومات الـ User اللي واصل بالـ Port .. (معلومات زي الـ MAC والـ IP).

والطريقة دي بتعتمد على ربط الـ Switch بـ VMPS Server زي الـ ISE .. والـ Server دا بيكون عليه بعناوين الـ MAC الخاصة بالـ End Devices .. وبالتالي أول ما جهاز معين يتوصّل بالسويتشر « يتم وضعه في الـ Vlan المناسبة على حسب الـ Configuration اللي انا عاملها.



« **Interface Instant Movability of end devices** يعني لو وصلت Client-A ناي هحط في الـ Vlan اللي انا عابرها

الطريقة دي مكلّعه وسخاچ Configuration معدهم

25.4. Types of Vlan

Default Vlan

هي الـ Vlan الافتراضيه الموجوده على اي Switch وسكون كل مياد السوسيس موحده بحها .. ولا يمكن حذفها او التعديل عليها

Data Vlan

هـ، VLAN عاديه لكن، سستخدمها في، سادل، البيانات الخاصه بالمسخدمين،

Management Vlan

Native Vlan

هي VLAN البرافيك سم ارساله من حلالها بدون Tag وسكون 1 VLAN ويمكن تعسرها .. وهنفهم الـ Tag كمان

Voice Vlan

هـ، سکه مخصوصه لیغا، الصوب و عط و عاره ساحد أولمهه و، الوجهه

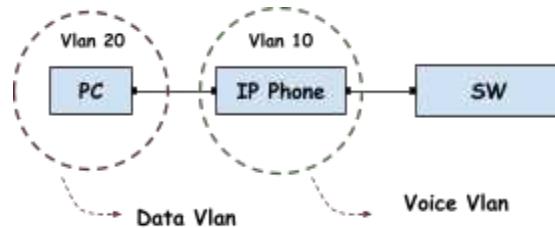
IP Phone II ، ۱۵

الـ IP Phone نجوي على اسـ Port .. واحد لوصـله بالـ IP .. والـ IP Phone يوصـل حـار آخر (PC) بالـ IP



طب انه سب اسخدام اس Ports، فى حن ان ممک نوصل الـ PC بالسويس معاشره:

- السب هو ليوفر معاشر على السويس
- لأن ميلا لو فى سويس يحتوى على عشره معاشر فقط
- وعدي 10 موظفين .. لكل واحد PC و IP Phone
- فى الحاله دى هحتاج 20 صعد على السويس .. فاما انى هحتاج ارود Rack على السويس لرباده المعاشر، او هوصل سويس بانى سه لكن الافضل والاوفر هو اسخدام IP Phone يحتوى على صعد .. وهنا ممک اعمل على السويس Voice Vlan حاصله بالمسخدم لمبرير البيانات الخاصه سه .. و IP Phone حاصله بالصوب لمبرير البيانات IP Phone بالـ
- سب اسخدام Vlan مختلفه لكل من الصوب والمسخدم
 - اولا للحمايه .. لأن المسخدم ممک ستصب على الصوب باسخدام بطيئات عبر مصرح بها
 - لو الاين فى نفس الـ Vlan
 - السب البالى ليطبق إعدادات حاصله على كل من الصوب والمسخدم .. لأن فى بعض المؤسسات بهمهم حدا أن الصوب تكون له أولويه فى المرور وعدم وجود Delay وبالالى يحصله فى Vlan مختلفه



يحصل ان الـ Voice Vlan واحد الـ ID من الـ Standard Range يعنى من 2 إلى 1001 عسان نصمم بوافعها مع أكبر عدد من الأجهزه فى السكه .. ولكن لو الـ Standard Range غير كافى « ممک سخدم الـ Extended Range ..

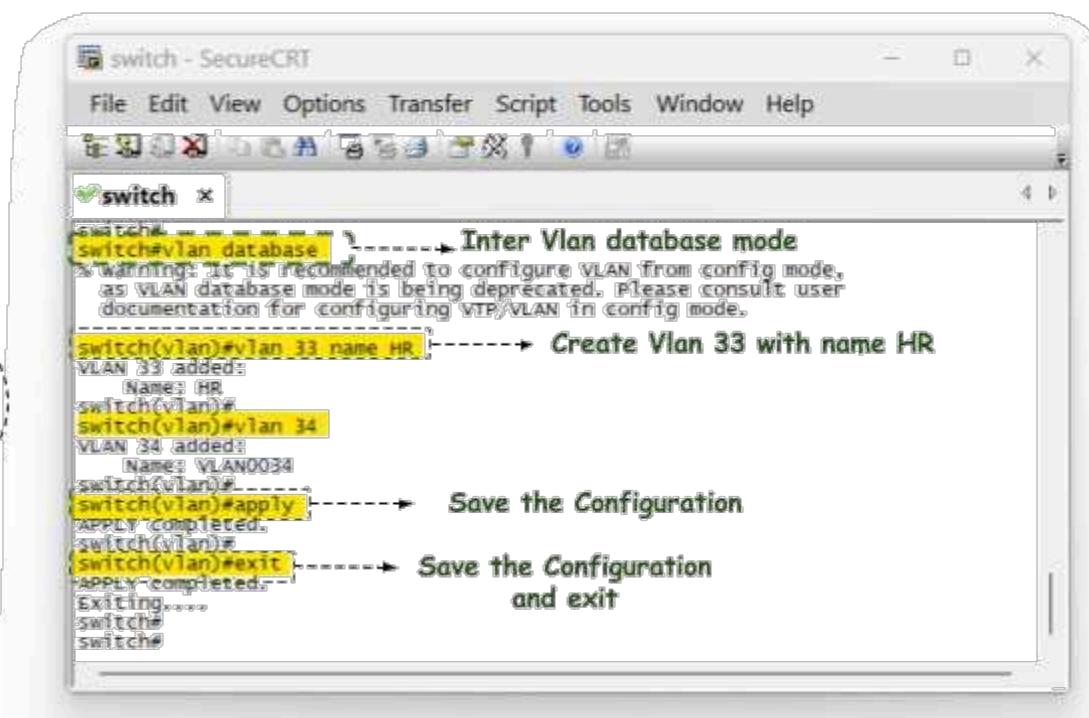
الـ IP Phone ممک واحد الـ Power او من حلائل بعنه الـ POE .. وممک واحد الـ Vlan من حلائل بروتوكول CDP او LLDP او من حلائل الـ Configuration DHCP Options

25.5. Vlan Configuration

في 3 طرق لعمل الـ *Vlan Configuration*

Vlan Database

الطريقة دي كانت شغالة على الـ Switches القديمة زي IOS Series 2590 او اي IOS قديم .. ومازال شغالة على الـ Switches الجديدة، لكن مش Recommended اتنا نشتغل بالطريقة دي



The screenshot shows a terminal window titled "switch - SecureCRT". The command "switch#Vlan database" is entered, which triggers a warning message: "Warning: it is recommended to configure VLAN from config mode, as VLAN database mode is being deprecated. Please consult user documentation for configuring VTP/VLAN in config mode." Below this, the command "switch(vlan)#vlan 33 name HR" is shown, followed by "VLAN 33 added: Name: HR". Then "switch(vlan)#vlan 34" is entered, followed by "VLAN 34 added: Name: VLAN0034". Finally, "switch(vlan)#apply" is run, resulting in "APPLY completed.". The sequence ends with "switch(vlan)#exit", "APPLY completed.", "Exiting....", "switch#", and "switch#". Annotations on the right side of the terminal window explain the steps:

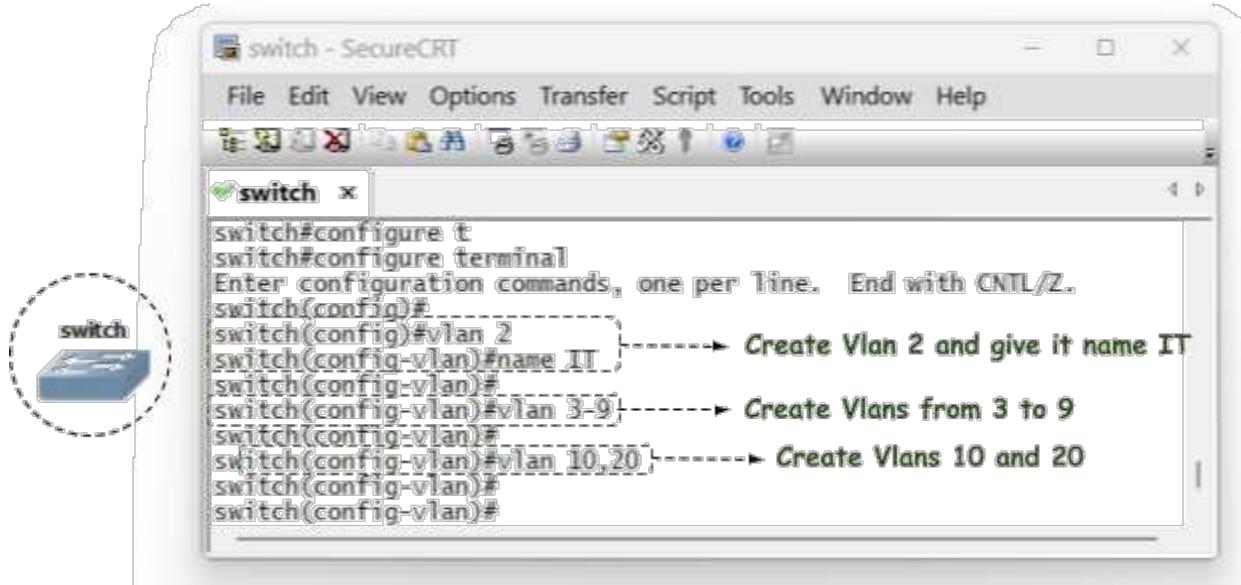
- Inter Vlan database mode
- Create Vlan 33 with name HR
- Save the Configuration
- Save the Configuration and exit

في الطريقة دي لازم اكتب *Apply* او *Exit* عشان الـ *Configuration* تتطبق

• الأمر *Apply* هيفضل في نفس الـ *Mode*

• الأمر *Exit* هيطلع للـ *Enable Mode*

Configuration Mode



لتطبيق الـ Vlan على Interface معين

```
switch(config)# int e0/1
switch(config-if)# switchport mode access
switch(config-if)# switchport access vlan 20
```

لتطبيق الـ Vlan على مجموعة من الـ Interfaces باستخدام أمر واحد

```
switch(config)# interface range e0/1 - 3
switch(config-if)# switchport mode access
switch(config-if)# switchport access vlan 40
% Access VLAN does not exist. Creating vlan 40
```

هلاحظ أنه حتى لو معمليتش Vlan 40 قبل كدا .. وطبقتها تحت الـ Interface <> هيعملها Create .. ودي

الطريقة الثالثة

للتحقق

VLAN Name	Status	Ports
1 default	active	Et0/0, Et1/0, Et1/1, Et1/2 Et1/3, Et2/0, Et2/1, Et2/2 Et2/3, Et3/0, Et3/1, Et3/2 Et3/3
2 IT	active	
3 VLAN0003	active	
4 VLAN0004	active	
5 VLAN0005	active	
6 VLAN0006	active	
7 VLAN0007	active	
8 VLAN0008	active	
9 VLAN0009	active	
10 VLAN0010	active	
20 VLAN0020	active	
33 HR	active	
34 VLAN0034	active	
40 VLAN0040	active	Et0/1, Et0/2, Et0/3
55 VLAN0055	active	
1002 fddi-default	act/unsup	
1003 token-ring-default	act/unsup	
VLAN Name	Status	Ports
1004 fddinet-default	act/unsup	
1005 trnet-default	act/unsup	

كل الـ Vlans اللي عملتها بتخزن في ملف `vlan.dat` داخل الـ FLASH

- لحذف `Vlan` معينة

```
switch(config)# no vlan 10
```

بعد حذف `Vlan 10` كل الـ Interfaces الموجودة فيها هيبقو Homeless ولون البورت هيكون

`Vlan` بس لو عملت الـ `Vlan` تاني الـ `Interfaces` هترجع لنفس الـ `Vlan` Amper

لحذف كل الـ `Vlan` Configuration على الـ `Switch` بنستخدم أمر `erase startup-config` •

بس الامر دا بيحذف الـ `Configuration` الموجودة في ملف `Startup-config` والـ `Vlans` بتبقى موجودة على الـ `Flash`.

لحذف الـ `Vlans` نهائيا .. استخدم أمر:

```
switch# delete vlan.dat
```

25.6. Vlan Port Types

أي Port على السويفتش، ممكن يكون Access أو Trunk

25.6.1. Access Port

"The Port belongs to only one Vlan"

يستخدم الـ Access Port لتوصيل السويفتش بالـ End Devices .. زي الـ PC والـ Printer والـ Server .. مش بيقبل غير واحده.

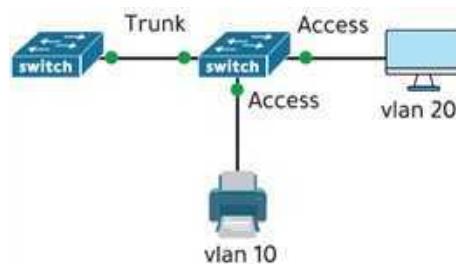
عشان كدا اي بورت متصل بـ End Device بـ Access عن طريق الأمر:

```
switch(config-if)# switchport mode access
```

25.6.2. Trunk Port

"The Port belongs to all Vlans"

يستخدم الـ Trunk Port لتوصيل السويفتش بـ سويفتش آخر .. عشان ينقل اكتر من Vlan بين السويفتشات



ليه عملو فكرة الـ Trunking

- لو في اتنين Switches متوصلين بعض وكل واحد عليه مجموعة من الـ VLANs
- لو عايز أخلي كل VLAN موجودة على SW1 توصل لنفس الـ VLAN المقابلة ليها على SW2 .. يعني لو تروح لـ 4 وهكذا .. نخلي الـ uplink اللي موصل السويفتشين في أي VLAN ؟



في حلين:

- اول حل هو توصيل السويفتلين باريحة Uplinks و كل VLAN في Uplinks معينة عشان يقدرو يصلو بعض .. لكن الحل دا مش منطقى وهىستهلك المنافذ على السويفتش و هيكلفني كابلات كتير.
- الحل الأفضل هو تحويل الـ Uplink اللي بينهم الى Trunk وطريقة عمله ان الـ Frame قبل ما يطلع من SW1 مثلاً « يتخطى عليها Tag او Sticker مكتوب عليها الـ VLAN اللي طالعة منها .. وبالتالي لو وصلت لـ SW2 هيبيعتها لنفس الـ VLAN .

25.7. DTP - Dynamic Trunking Protocol

الـ Default على السويفتش انه بيعمل Negotiation مع السويفتش المقابل عشان يحدد نوع البورت Access ولا Trunk .. ودا بيحصل عن طريق بروتوكول الـ DTP الخاص بـ Cisco .
بروتوكول DTP بيكون مفعل By Default .. وأول ما السويفتش يستقبل DTP « بيحول البورت لـ Trunk .. او بيحوله لـ Access لو مستلمش DTP .

في الحالتين للـ Dynamic Trunking :

- Dynamic Auto يعني السويفتش بيقل الـ Negotiation بس منظر DTP من البورت المقابل
- Dynamic Desirable يعني السويفتش هو اللي بيبدأ التواصل مع الطرف الثاني وبيجعut DTP

وفقاً للحالتين دول، ممكن نستنتج أن:

- لو الاثنين Interface المقابلين لبعض « حالتهما Dynamic Auto ، كل واحد هيكون منظر DTP ومتش هيحصل Trunking
- أما لو الاثنين أو واحد منهم Dynamic Desirable هيحصل Trunking

بعض السويتشات بيكون الـ Default عليها Series 2590، 2560 زي Series 3550 زي Desirable.

لمعرفة حالة الـ Interface

```
switch# show int e0/0 switchport
Name: Et0/0
Switchport: Enabled
Administrative Mode: dynamic auto
Operational Mode: static access
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
```

Default or applied mode
Result of negotiation
DTP On
support ISL and dot1Q

Administrative Mode: dynamic auto

الـ Default Mode للبورت معمول "dynamic auto"، يعني البورت مستنى الجهاز اللي متوصلك بيها هو اللي يطلب إنه يبقى Trunk.

Operational Mode: static access

البورت حالياً شغال "static access" ، بسبب انه مستنى DTP عشان يبقى Trunk.

Administrative Trunking Encapsulation: negotiate

نوع الـ trunk encapsulation (سواء كان IEEE 802.1Q أو ISL) متظبط إنه يتفاوض عليه، بس مفيش تفاوض هيحصل لأن البورت مش شغال Trunk أصلًا.

Negotiation on trunk on

معناها ان بروتوكول الـ DTP شغال

بروتوكول DTP بيتم إرساله كل 30 ثانية .. حتى لو خليت حالة البورت Trunk وبالتالي لازم اعمله لو عايز اوقف رسائل الـ DTP.

لعمل DTP لبروتوكول الا

```
switch(config-if)# switchport nonegotiate
```

لإعادة تشغيل الا DTP

```
switch(config-if)# no switchport nonegotiate
```

لعرض معلومات عن بروتوكول الا DTP

```
switch# show
switch# show dtp
Global DTP information
    Sending DTP Hello packets every 30 seconds
    Dynamic Trunk timeout is 300 seconds
    16 interfaces using DTP
```

- لاحظ ان الا DTP Frames تتبع كل 30 ثانية عشان تفضل عملية الا Trunk شغالة بين السوينتشات.
- لو البورت مستقبلش DTP Frames من السوينتش المقابل خلال 300 ثانية او 5 دقائق، السوينتش هيوقف الا Trunking على البورت دا .. وهيسقبل بيانات من الا Native Vlan فقط .. إلا لو محدد الا Trunking بشكل يدوي.

25.8. Vlan Trunking Protocols

شركة Cisco هي أول من طور فكرة الـ Trunking، وبعد حين بدأت شركات تانية في تطويره .. وبيستخدم عشان السويتش يعرف الـ Frames اللي تم استلامها جاية من اي Vlan وفي أربع أنواع من الـ Trunking Protocols .. اتنين منهم شغالين على Cisco.

Inter-Switch Link - ISL

هو بروتوكول قديم مملوك لشركة CISCO ولا يعمل على الأجهزة الحديثة

- يعمل على Frame Encapsulation
- مش بيدعم الـ Native Vlan

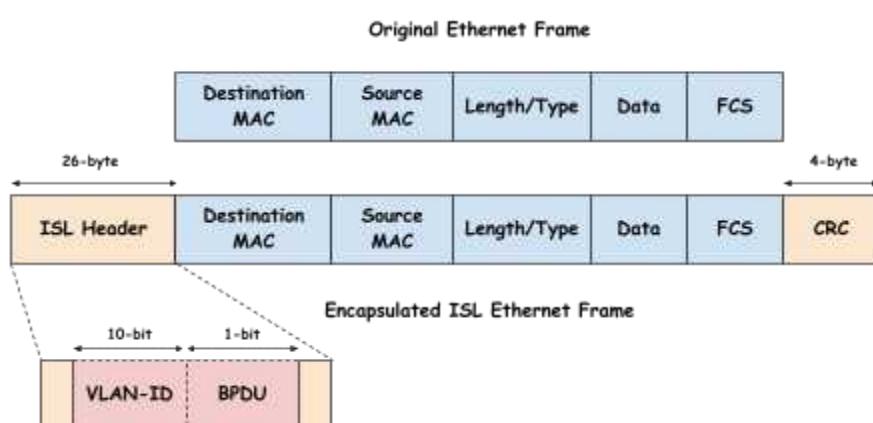
802.1Q (Also called dot1Q)

هو Standard Protocol طورته منظمة IEEE .. ويشتغل على أجهزة Cisco وغيرها

- يعمل على Frame Encapsulation
- بيضيف له Tag بتوضح الـ Vlan
- بيدعم الـ Native Vlan

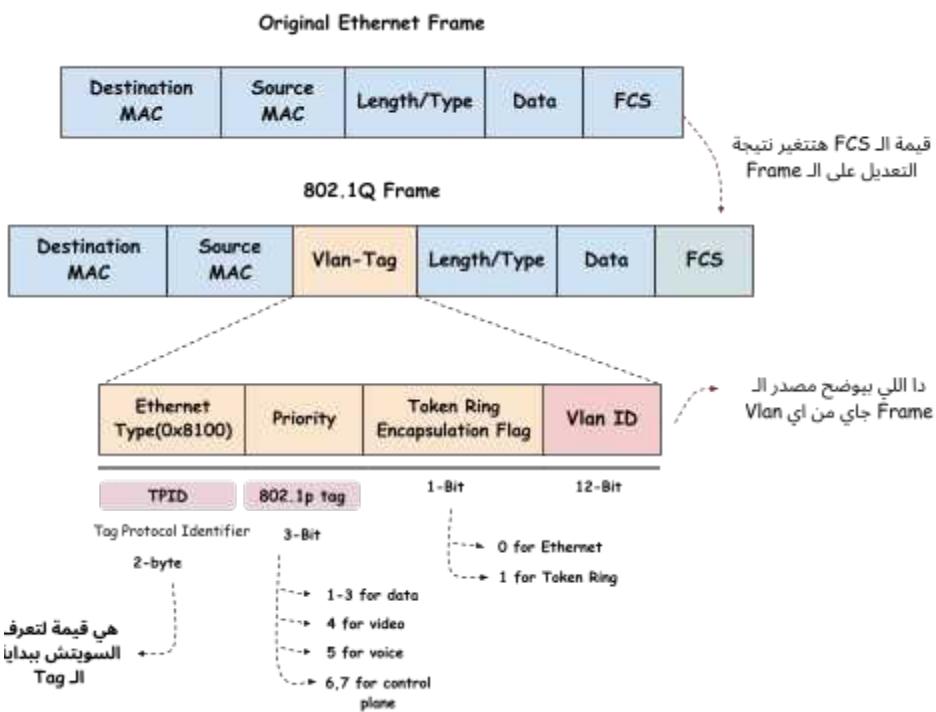
الفرق بين ISL و dot1Q

- بروتوكول ISL بيضيف Header و Trailer على الـ Ethernet Frame على الأصلية ..
- والزيادة دي بتقلل الـ Performance بتاع الشبكة



الـ ISL Header فيه معلومات كتير .. واهم حجة فيه هو الـ Vlan-ID والـ BPDU

- أما بروتوكول dot1q فيزود Tag فقط بين الـ Source MAC والـ Destination MAC



ميزة استخدام الـ Native Vlan ان السوينتش مش هيضيف Tag على الـ Frame لو جاي من الـ Native Vlan. وبالتالي هيوفر أربعة Byte لكل Frame .. ودا هيساعد في تقليل المعالجة وتحسين الأداء خصوصاً مع استخدامها للـ Voice over IP، لأن كل بايت يفرق في الجودة وتقليل التأخير.

لتعديل الـ Native Vlan

```
switch(config)# int e0/1
switch(config-if)# switchport trunk native vlan 20
```

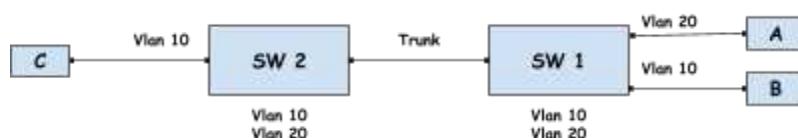
لو عدلت الـ Native Vlan على سوينتش، ومعدلتهاش على السوينتش المقابل <> هيطلع syslog كل 60 ثانية بتقول لازم اعدلها .. ودا بيحصل بسبب بروتوكول الـ CDP.

25.9. Allowed Vlans

ممكن نسمح او نمنع مرور الـ Traffic الخاص بـ Vlans محددة بين السويتشات لو الـ Interface اللي بينهم . Allowed Vlans تكون عن طريق خاصية الـ Allowed Vlans .. الإعداد الافتراضي ان كل الـ Trunk تكون

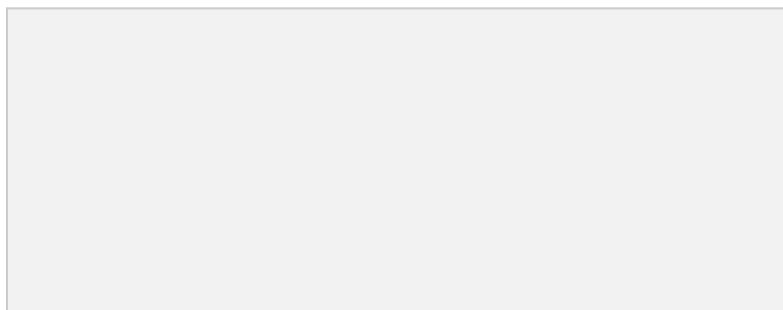
طب ليه ممكن نمنع مرور بعض الـ Vlans !

في الـ Topology اللي في الصورة .. لو جهاز B اللي في Vlan 20 بعث Broadcast «» الرسالة هتوصل لـ SW1 فهيبعتها على كل الـ Interfaces اللي وصلت منه .. وبعدين هتوصل لـ SW2 وهيعملها Drop بسبب ان مفيش اجهزة عندها في Vlan 20.

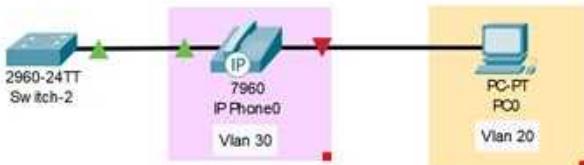


وبالتالي ممكن نقل الـ Broadcast Domain عن طريق تحديد الـ Vlans اللي فيها أجهزة على السويتشين فقط .. يعني في السيناريو دا .. هحدد على SW1 انه ميعتش بيانات من Vlan 20 عن طريق ازالتها من الـ Allowed vlans.

في سيناريو آخر .. لو عندي SW3 متصل بـ SW2 وعليه جهاز في Vlan 20 «» مينفعش اشيل Vlan 20 من الـ SW1 على عشان جهاز D يقدر يصل لجهاز A.



25.10. Voice vlan configuration



```
Switch# conf t
Switch(config)# int f0/3
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 20
% Access VLAN does not exist. Creating vlan 20
Switch(config-if)# switchport voice vlan 30
% Voice VLAN does not exist. Creating vlan 30
```

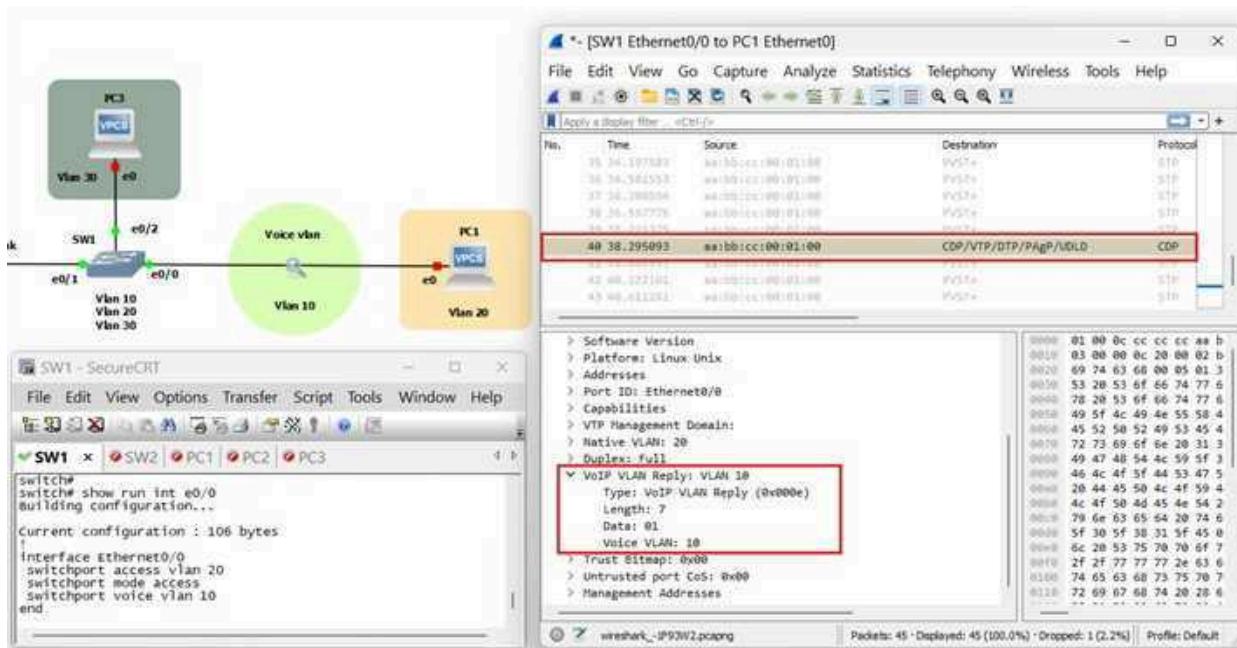
دي بعض الـ Options .. لكنها مس موجوده على الـ Packet tracer

```
switch(config-if)# switchport voice vlan ?
<1-4094> Vlan for voice traffic
dot1p Priority tagged on PVID
name Set VLAN when interface is in access mode
none Don't tell telephone about voice vlan
untagged Untagged on PVID
```

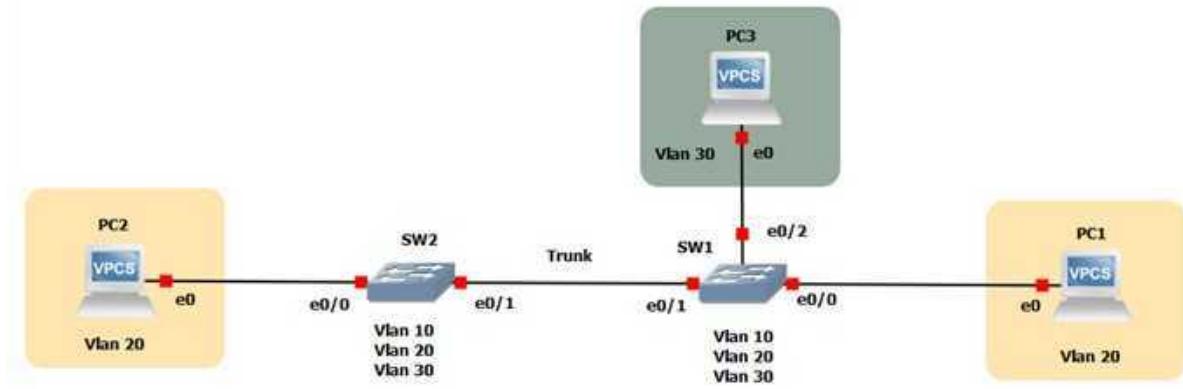
- ممک احدد Vlan معنه لـ Voice مناسره
 - الأمر name لتحديد Vlan معنه ساء على اسمها
 - الأمر dot1p هیحلی السویس نعمل Advertise لـ IP (ناسخدام بروتوكول ری الـ CDP او CoS Value مع Vlan ID 0 <> يعني نستخدم 802.1p Frame (LLDP)
 - الأمر none هیحلی السویس سمح لـ IP Phone ناسخدام اعداداته الحاشه
 - الأمر untagged هیحلی السویس نستخدم بروتوكول الـ CDP او LLDP لاعلام الـ IP Phone بإرسال Unagged البيانات

الـ Forward Frames على حسب الـ Vlan ID او دـ 0 .. السووسـ سعملها Untagged لـ وصلـ للسووسـ Configured Access Vlan المـوحـدـ على الـبورـ اللـى وصلـ مـهـ او من حـالـ الـ Native Vlan لـ معـسـ Access Vlan

لو طبعنا الاب على الـ GNS3 وملأه هنالك ان الـ Traffic لـ Capture سمع من حلال بروتوكول الـ CDP.



Allowed Vlan Configuration



هنالاحظ في الاب دا ان في Vlan 10 و 20 و 30 على كل من sw1 و sw2 .. لكن sw1 بس هي اللي فيها اجهزة على السويفتين .. وبالتالي هنسمح بال Traffic من 20 Vlan فقط

```
sw1(config)# int e0/1
sw1(config-if)# switchport trunk allowed vlan ?
WORD      VLAN IDs of the allowed VLANs when this port is in trunking mode
add       add VLANs to the current list
all       all VLANs
except    all VLANs except the following
none      no VLANs
remove    remove VLANs from the current list
```

ممكن نحدد أرقام ال Vlans مباشرة عشان نسمح بيها .. وممكن نعمل add لـ Vlan للقائمة الحالية من ال allowed VLANs .. او all للسماح بكل ال Vlans .. او except للسماح بكل ال Vlans ما عدا معيينة .. او none لإلغاء مرور كل ال Vlans

اول حاجة .. نحدد نوع ال Trunk Encapsulation وبعدين نحوال نوع البورت الى Trunk <> ثم نحدد ال Allowed Vlans

```
sw1(config)# int e0/1
sw1(config-if)# switchport trunk encapsulation dot1q
sw1(config-if)# switchport mode trunk
sw1(config-if)# switchport trunk allowed vlan 20
```

Verifying

```
switch# show interface e0/0 switchport
Name: Et0/0
Switchport: Enabled
Administrative Mode: static access
Operational Mode: static access
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 20 (VLAN0020)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: 10 (VLAN0010)
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk associations: none
Administrative private-vlan trunk mappings: none
Operational private-vlan: none
Trunking VLANs Enabled: ALL
Pruning VLANs Enabled: 2-1001
Capture Mode Disabled
Capture VLANs Allowed: ALL

Protected: false
Appliance trust: none
```